JPRS-UCC-90-007
23 OCTOBER 1990

FOREIGN
BROADCAST
INFORMATION
SERVICE

JPRS Report

Science & Technology

USSR: Computers

19981217 121

Science & Technology
USSR: Computers

JPRS-UCC-90-007 — 23 OCTOBER 1990

CONTENTS

GENERAL

SOFTWARE

APPLICATIONS

The Key to Acceleration

907G0040A Moscow SOTSIALISTICHESKAYA INDUSTRIYA in Russian No 276, 2 Dec 89 p 2

[Article by V. Volnov]

[Text] "If you were to work for me" we learn from Dr. Knute Krumach "I would have to pay each of you a thousand marks a day. It is remarkable that you have succeeded in creating such a high quality system in such a short time period."

The president of the West German company "NS-SC" had good reason to be surprised. The company was displaying its products at one of the international exhibits in Moscow. These products included computers, software, and other modern electronic equipment. It was not advantageous to ship the exhibits home and the company delivered the computers to Soviet specialists under mutually agreeable terms.

"When this technology became available to me," relates V. Tyurin, director of the creative research team of the self-supported scientific-technical center at the "Save the World and Nature" All-Union association, "I realized that our hour had arrived." The technical capabilities of the computers made it possible to implement our most ambitious designs. We established a goal for ourselves: create software that is capable of competing on the international market. It is no secret that software today is a desirable and very expensive product on the international market. Software prices are constantly rising and are a very marketable product. Software development itself has become a unique art. Thousands of companies compete in this area world-wide and a fair amount of experience and training are required to participate in this competition.

"We took advantage of the Japanese experience," says V. Tyurin. "We initially collected information on all available programs throughout the world in our fields on interests. We analyzed their weak spots and tried to determine what we could add to these programs. One thought did occur: we decided not to simply write a single program but rather develop a system that would be able to churn out programs like an automated assembly line. Of course, reflecting the individual customer taste and requirements."

The word "work" does not fully reflect the state in which V. Tyurin and his "team"—senior scientific associates A. Zakrevskiy and Yu. Bukhteyev—worked for over six months. They sat behind computer consoles until they were seeing green circles. However, after six months they developed a unique automated system for producing control programs for numerical program control systems. According to specialists, under ordinary circumstances it would have taken years and years to develop this system.

Why work at such a frantic pace? If you wish, to demonstrate to ourselves that it was possible. They also proved this to themselves and others: we can develop exceedingly complex scientific goods for the international market.

"Today no matter where you look," says Tyurin, "you hear one thing: we can't do anything, we'll never catch the west. Such conversations do little other than irritate me and prompt me to protest: what, are we dumber than everyone else in the world? How long can you go on complaining? It's time to act!"

And act the West German businessman did. He signed a contract to purchase the system. Hard currency is the best indicator of the ability of goods to compete. Negotiations are ongoing today to sell the system in Sweden, England, Austria, and China. What are the advantages of this design over analogous foreign designs?

The automated system can be used to produce a variety of programs at a very high rate of speed: 2-10 times faster than before. An engineer will spend no more than 15 minutes in developing a moderately complex program. We note that there are programs that would require under ordinary conditions several months and even years. Clearly any reduction in time in this area will generate enormous cost savings.

However the advantages of the new design are not limited to this area. The system is universal. It can be used to generate programs for drilling, cutting, milling, and electronic erosion manufacturing sets and for machining parts of any shape. With such a system enterprises can substantially reduce the time from idea to implementation. This is particularly important in the case of small-scale and flexible manufacturing conditions. The idea itself is not the only original aspect: the "packaging" proposed by the designers was also original.

"We incorporate all customer-specified modifications in the design and provide a one year guarantee and at the end of this year we provide an updated system at no charge," says Tyurin. There are many other additional services. In order to break into the international market it is necessary to learn to live by its rules. One of these rules is that a new product must have the newest user qualities.

Yet isn't it impossible to work continuously at such a pace as you have done?

"Today this is not only possible but is in fact necessary," my interviewee notes with some confidence. "This, of course, is true if we want to take an appropriate place among the developed nations. Judging both by my experience and that of my colleagues, specialists are yearning to work at full speed. If all the stored up energy were to be released it is entirely possible that the world would speak of the 'Russian miracle.' For example, even today we are prepared to cooperate with any interested partner both domestically and abroad."

These words are not a bluff. Proof of this can be found in the "Elektronika-2000" International Exhibition which

recently was sponsored by the American-Polish company IDM at the USSR Exhibition of National Economic Achievements; this company showed products at the exhibit containing software developed by Tyurin's team. And this was the only exhibit of domestic, Soviet origin.

Inoculate the Computer

907G0039A Moscow SOTSIALISTICHESKAYA INDUSTRIYA in Russian No 281, 9 Dec 89 p 4

[Article by O. Vladimirov]

[Text] Moldavian specialists propose an original method of dealing with electronic "viruses": insidious programs that strike computer memories.

The display lights up with letters and numbers. However, suddenly the ordinary, flawless picture changes. The characters blur together and begin to fall like rain droplets. After a few minutes they congeal into puddles of light...

"This is how a typical computer virus operates," explains A. Moldovyanu, a leading specialist at the Kishinev Automated Management System Design Bureau. "We encountered such an illness for the first time in the middle of last summer. Then it destroyed many months of work: an entire program package. Then each month we had new surprises. Another tragic coincidence was that the electronic virus arrived in Moldavia at the same time as AIDS.

There are more than 100 computer viruses throughout the world today. They have different symptoms. Sometimes a jumping ball of light will appear on the screen, sometimes letters will begin to flicker or lines will shift and occasionally all information will suddenly swirl into a soupy mess, confusing and flabbergasting the operator. And if the first viruses produced by talented, mischievous and unknown programmers were comical and harmless, they have since acquired aggressive, brutal characteristics. Major western computer and software manufacturers have begun to use viruses in their internecine struggles and competition. Special programs are under development in secret laboratories. Once they enter ordinary working programs, just like biological viruses, these programs spread among magnetic disks and tapes with the circulated software. They then wait for the appropriate hour. The authors of the virus programs have inserted a certain date to activate the virus. After the incubation period the virus, like a time bomb, can simultaneously destroy all information on hundreds of thousands and even millions of computers. The damage caused by this electronic epidemic is growing daily.

Computers in this country have been struck as well. For example, we know the story at one Volga automobile factory. A fired programmer generated noise on the computer in revenge and the factory suffered serious losses. This was a rather simple case as the source of the infection was known. The case in Moldavia was more complex. The "virus" struck an automated management system design bureau. The epidemic then spread to other bureaus, institutes, enterprises, and organizations. Its specific origin remains unknown.

"Our efforts to deal with electronic illnesses have clearly been insufficient so far" notes Aleksandr Andreyevich. Mostly preventive measures are used for computer protection: ordinarily a different program is tested on one computer and will only then run on all other computers. We have developed a unique vaccine. If this vaccine can be used it will be possible to generate something similar to immunity in computers."

The new universal protection system has been called "Kobra" [Russian acronym: operational security system]. Nine months were invested in developing this system. The system consists of several programs that, unlike their predecessors search out and destroy any viruses. This includes both existing viruses and any theoretically possible viruses. The "Kobra" system has already demonstrated its effectiveness at one of the major instrument-building enterprises: the Kishinev "Mezon" factory. Here the system is used to protect the automated operator workstations. Perhaps other organizations will also show a strong interest in this new innovation.

"There is interest" according to Aleksandr Andreyevich, "even abroad, although this will be of little help to us. Our country is not a signatory to the International Convention on Software Patent Rights. This is unfortunate. The entire world highly values software since even the most advanced computer is useless without software. Last year the US sold 24 billion dollars worth of computer programs on the international market. And of course we have no fewer talented programmers. We could also sell programs, including the 'Kobra' system. But how can we do this if we still don't know who owns intellectual property? It is necessary to make a compromise. Western companies "import brain power" and utilize our inexpensive brain power to set up joint ventures for computer programming. A similar situation exists with the 'Kobra' system. I fear that we will lose our advantage here as well."

Mechanization and Automatization of Control: Industrial Research Collection

907G0150A Kiev MEKHANIZATSIYA I AVTOMATIZATSIYA UPRAVLENIYA: NAUCHNO-PROIZVODSTVENNYY SBORNIK in Russian No 1, Jan-Mar 1990 pp 62-64

[Abstracts of collection of research works]

[Text]**ABSTRACTS**

UDC 621.317:681.31

Organization of a Self-teaching System to Search for Manufacturing Defects in Electronics Units. N. P. Bayda, V. I. Mesyura

It is proposed that self-teaching be used to increase the effectiveness of a new class of diagnostics systems: manufacturing defect analyzers. Automatic decomposition operations upon the model of the standard reference electronics unit yielding two-pole and multiple-pole components corresponding to resistors, capacitors, p-n junctions, etc., are accomplished to implement the self-teaching procedure. The obtained data are used to generate automatically a sequence of test operators of the working program to diagnose electronic units. The introduction of the self-teaching process for manufacturing defect analyzers enables one to accelerate the diagnostics program preparation process by a factor of four to six. Two illustrations, three references.

UDC 658.012

Models to Increase the Effectiveness of Introducing Scientific and Technical Progress Measures in Conditions of Self-financing. A. A. Panchenko, A. V. Sadovskiy, Ye. V. Matyush

The inclusion of the task of the optimum allocation of production, science and engineering development funds for the introduction of scientific and technical progress into automated management systems at enterprises under conditions of self-financing enables one to significantly increase the effectiveness of the funds' utilization and provide a profit for the enterprise. Three references.

UDC 658.310

Evaluation of the Level of Work Mechanization and Automatization. V. A. Dedekayev

The energy approach to measuring the level of work mechanization and automatization, which is defined as the ratio of the machine tool energy costs for executing a manufacturing operation to total costs (machine and human), is implemented in the utilization of an electronic measuring complex with analog to digital data converter and data transmission by a serial code over a radio channel to a receiver coupled with the Pravets-16 computer (PK-YeS 1839). One table.

UDC 002.55:681.3

Information Survey of Enterprises. V. Ye. Khodakov, V. I. Perederiy

The determination of the quantities of information regarding a preselected basic product is proposed for the information survey of enterprises. This enables one to simplify and accelerate the creation of automated management systems. Two illustrations.

UDC 658.5.012.4.011.56

Use of Recursion to Navigate Through a Hierarchical Data Base. O. N. Sleptsova

The use of a recursive procedure to navigate through a hierarchical data base frees the programmer of the necessity of the exact knowledge of its structure and enables one to reduce the programming costs and increase the effectiveness of the working program. The procedure can be used in the transition from one data base management system to another, the creation of a reserve copy, when restructuring the data base, etc. One illustration, four references.

UDC 681.322.067

Automation of the Design of the Conceptual Schema of the Subject Field with the Application of an Expert System. A. I. Deshko

During the conceptual design, the problem of constructing the schema of the model that integrates representations of the information requirements of all user classes is solved. The automation of the conceptual design is provided by the application of an expert system whose knowledge base contains information regarding all components of the utilized conceptual model, the design strategy that specifies the order of description of the subject field fragments, and the rules of their integration into the global schema. Four references.

UDC 621.391

Informatization: Factological Modeling of Development Processes on the Basis of Knowledge. V. Ya. Ruban, V. V. Vasilyev, O. V. Ruban

The restructuring of all spheres of social activity accomplished in the country requires the use of a broad spectrum of knowledge, without which the implementation of the solutions can result in unpredictable and irreversible consequences. The informatization of Soviet society is called upon to provide the means, adequate to the needs, of representing the facts of reality as primary knowledge, the logical deduction of new knowledge, and the effective access to them by various types of users. Two tables, four references.

UDC 681.324

Set of Programs for Effective Calendar Planning of Computational Work in an Industry Computer Center. Yu. P. Zaychenko, D. M. Vishtal, A. Ye. Pakrysh

The application of a program set for effective calendar planning enables one to automate the compilation of computational work execution plans at computer centers. As a result, the computer center losses are diminished as a result of reducing penalties for missed deadlines in the execution of user tasks. Three references.

UDC 681.3:658.52.011.56

Effective Planning Module for a Flexible Automated Bay with Built-in Simulation Model. A. A. Lavrov, L. S. Yampolskiy, S. L. Yampolskiy

The use of simulation modeling units in the control system of a flexible production process provides the combination of a more precise description of the object with the capability of automated decision making and effective matching with the operation of the other system elements. The effective planning module implemented on the basis of the simulation approach is a composite part of an interactive system for effective planning and situation control of the "Dispatcher-1" flexible automated bay developed at the Kiev Polytechnical Institute. Two illustrations.

UDC 681.326.75:621.643

Effective Identification of the Technological Characteristics of the Main Lines in an Automated System to Control Technological Processes. A. N. Parashchak

Kalman filtering theory is used to develop a recursive algorithm to identify the technological characteristics of the main lines for an automated technological control process to transport petroleum. The recursive algorithm is most efficient for real time control systems, since its application creates the prerequisites for optimization of the computational process. Two references.

UDC 681.3:62-52

System to Control Transportation Operations for Automated Warehouses. A. G. Gorbunov, S. M. Baronov, A. R. Pavlenko

A microprocessor system to control transportation operations based on the DVK- 2 microcomputer with I2 15KS-180-032 parallel exchange apparatus is used at automated warehouses to execute loading and unloading operations of any duration. One illustration, two references.

UDC 681.3:616.015

Computer Monitoring of Drug Therapy Safety. V. Ye. Kalugin, M. K. Dudareva, V. A. Dutchak, P. A. Angelutsa, A. P. Viktorov

A system for the computer monitoring of drug therapy safety is based on the information technology of logical decision making that implements deductions to prevent complications in drug therapy and undesirable interactions of simultaneously utilized drugs. It can be used in hospitals, clinics, health facilities, and military hospitals, and as a teaching system in medical institutes and in courses to increase the skill level of physicians.

UDC 002.5:330.341.1

Information Technology to Generate Scientific and Engineering Programs in Construction. B. A. Voloboyev, A. L. Ilnitskiy, Ye. V. Grisha, G. L. Pomirchiy

An automated system to monitor the implementation of scientific and engineering programs has been developed and is in use at the Scientific Research Institute of Automated Planning and Control Systems for Construction of the Ukrainian State Construction Ministry. This system does not possess specific peculiarities and can be used by organizations during the prospective planning of production development. One illustration, five references.

UDC 519.68

Development of a System for the Intelligent Generation of Scientific and Engineering Programs in the Ukrainian Republic Automated System to Manage Science and Technology on the Basis of Knowledge Bases. V. M. Chmyshenko, G. A. Putilova

The Ukrainian Republic Automated System to Manage Science and Technology has developed a system based on knowledge bases for the intelligent generation of scientific and engineering programs in accordance with contemporary artificial intelligence paradigms. Six references.

UDC 621.3.035.2:658.562

Apparatus to Control the Quality of Objects Made of Carbon. Yu. A. Medvedkov, A. N. Tupikov, T. G. Shakhova, V. I. Bondarchuk, N. V. Negutorov

The Zaporozhye branch of the Tsvetmetavtomatika All-Union Scientific Research Institute of Computer Science and the State Scientific Research Institute of the Electrode Industry have developed an apparatus to control the quality of carbon objects by measuring the elastic vibration propagation time by the shock-wave method. The apparatus can be used to control objects and large-scale structural materials such as concrete, ceramics, refractory materials and rock. The annual savings from the introduction of the apparatus comprised 18 thousand rubles. Two illustrations, four references.

UDC 692.415.002

Unified System to Control Roofing and Finishing Machines. V. I. Vorobyev, A. G. Samoylenko

Firmware is used to create a unified system to control roofing and finishing machines. Tools based on the SM-1420 macroassembler were developed to automate the programming of the read-only memories. One illustration, five references.

UDC 681.3

The Optan-mikro Personal Image Processing Station. A. Ye. Lysenko, A. V. Stolyarenko, S. V. Polishchuk, I. A. Fursik, V. M. Kozub

The personal image processing station was designed to execute the principle types of image processing and analysis operations at the perception rate of the operator. The station was implemented on the basis of the IBM PC

personal computer and an image processing unit. Three illustrations, three references.

UDC 658.5.012-011.56:664.6/7

Mathematical Model to Control Occupational Safety. A. I. Kharitonov

The use of the matrix method of estimating the occupational safety and comfort level enables one to construct a mathematical model for controlling the occupational safety system at various industrial sites. Two references.

UDC 519.68

Experience of Constructing the "Resource Preservation" Knowledge Base by Logic Programming Tools. G. A. Putilova

The Main Scientific Research Computer Center Adminstration of the Ukrainian Gosplan has produced the "Resource Preservation" knowledge base, which contains a description of material resource economy factors and means of resource conservation, and their dependencies on the specific features of individual kinds of resources. The knowledge base records are written in a language close to natural language in the form of logical formulae, and can be dynamically augmented or modified.

UDC 621.395

Control of the Operating Condition of Fiber-optical Transmission Lines and Localizing their Faults. B. P. Borisov

When organizing the maintenance of transmission lines of any transmission medium, one must be guided by a fundamental principle: the availability factor, mean time between failures and the mean service restoration time must lie within prescribed bounds for the entire service life of the transmission line, which stipulates the necessity of automating the maintenance.

The automated maintenance system includes telemetry apparatus and a controller or a data acquisition and processing unit. Two tables.

UDC 681.3:518

Evaluation of the Effectiveness Characteristics of Technological Information Processes in an Automated Enterprise Control System. A. A. Belous, V. V. Rachek

The three-step procedure to design technological information processes for data processing in an integrated automated enterprise control system provides the required effectiveness characteristics for supplying and processing data. The evaluation of the effectiveness characteristics is accomplished on the basis of a developed mathematical model which enables one to consider the time required to transmit the processing request, process the requests in the computer, and transmit the response from to the computer to the remote user. Two illustrations.

We Must Not Give Up

907G0124A Moscow TEKHNIKA I NAUKA in Russian No 2, Feb 1990 pp 4-6

[Interview with Boris Leontyevich Tolstykh, Chairman of the USSR State Committee on Computers and Data Processing]

[Text] [Question] At the outset, could you tell us something about the committee which you head, its functions, and who is working here?

[Answer] In my understanding, the prime goal of the Committee today is to work out an effective government policy in the field of computerization of Soviet society. We must ascertain and scientifically validate the directions and methods so that the growth of our country takes place on the level of the advanced countries in the community of nations. After all, backwardness could result in our speaking a language different from that of the developed world. Therefore, the formation of a quality state policy requires very painstaking scientific analysis. And not just a glance by office workers or individuals, but a large collective project of scientists and specialists in both the sector of development and that of production. There is a common belief that one should not worry about a particular policy, a long range plan, but should tackle the immediate practical problems. I cannot agree with this. Unfortunately, we must admit that at present our country lacks a clearly formulated policy for computerization of society. We are now involving increasing numbers of scientists and specialists so that at least a first, tentative version can be worked out. I even think that it will not, can not, be the ideal version. Life does not stand still. Worldwide trends change, new engineering achievements appear, there is a new understanding of the various problems. If there is something that we do not yet know or understand, let us leave that part blank, but it is essential to produce a first draft now, and later continue to work out a new one, which should be replaced in due course.

Today, the Committee has a little more than four hundred colleagues. I have made no major replacements, we are still getting to know one another. I cannot say that we lack qualified staff, I feel that there are smart and responsible people here, but not enough time has passed for a serious assessment. Perhaps this period of familiarization will last till the end of the year. After all, the work is not just taking place in the Committee—it is necessary to get to know the staff of the ministries, the other enterprises. As the matter proceeds, I will also become acquainted with the techniques, methods, and results of the activity of the central bureaucracy. Let me be candid—there are many disagreements in our understanding of how to solve this or that problem, of how to cooperate. I do not wish to say that I am in possession of the absolute truth, only that we are not yet a unanimous

team, we have not yet worked out all lines of communication, and thus there are still many unwarranted difficulties, many mistakes. At the present stage, we are operating on the principle of open discussion, we are holding working meetings. The groups of top-priority problems have already been identified. We are discussing them and seeing what everyone thinks, giving them a chance to air their opinions. At first, we are calling upon the central bureaucracy, and when we feel the time is ripe to make a decision, we will involve the scientific institutions of the ministries.

[Question] Computer technology is a crucial factor determining the level of scientific and engineering progress. I dare say that, for our country in particular, this factor is even more crucial than the prestigious space research. How do you size up the situation in this area, under two aspects, first, as compared to the average level of achievement in the world, and second, in the long range view, allowing for the production and science capacity and the training of employees?

[Answer] The growth of computer technology has been made possible by the development of electronics, and the "informatization" of society is possible only on a basis of computerization. Therefore, microelectronics and computer equipment are naturally of paramount importance. The computers support the growth of informatization, they perform the calculations, control the processes, automate the processing of information flows. For informatization is the organization of global, massive flows of information, information services of the most diverse kind—from the simplest directories, like a train schedule, to commercial services, automation of the labor of the designer, the scientist, and the dissemination of the latest knowledge.

[Question] One way or another, there is a direct relationship—the effectiveness of informatization will be greater as the level of development of computer technology increases.

[Answer] Such a relationship does exist. The revolutionary significance of the personal computer is that it has made possible an individual style of work, savings of time, and a boost in labor productivity. But today this is only part of the problem, viewed from the standpoint of the end result. Another, no less important part of the problem today is to facilitate interaction of the personal computer and the person using it, not only with his own local and closed database, but with an entire network. If I have a computer, if you have a computer and we are joined in a network, we have a double advantage, and if there are not just us two, but many people with different interests, in possession of different knowledge, if our computers are hooked up to a single network and have access to a mainframe, holding an enormous bank of data, and if we can obtain any piece of information of interest to us in the space of a few seconds, imagine how much we gain! That is the crux of the matter. And the other neighboring fields—switching equipment, various forms of communication—have also achieved such level of development as makes it possible to integrate all this hardware and supply any information to distant subscribers almost instantaneously: within a region, within an immense country, within the international community. Imagine if, at this very minute, you could instantly access a data bank in any given country, one holding all the information you are interested in, for example, on the latest technology or on a method of treating diseases. And any other country could just as quickly obtain any piece of information kept in your data bank. There is great interest in a mutual exchange of information. And in many cities around the world there are already so-called "teleports," large complexes which use the latest equipment to organize tremendous flows of the most diverse information. This type of task can only be handled by superintegrated systems. Do you think a single computer could accomplish it?

[Question] Surely not, yet how close are we to what you are talking about?

[Answer] Of course, we are only at the very start of the journey. Our backwardness is very great. There are general problems which perestroyka has revealed—we do not know how to work, and the economic mechanism is imperfect. Today, the state must direct its resources to correcting such critical and primary problems as feeding the nation and providing consumer goods. Under these conditions, very many extremely difficult problems arise. Our sector needs more outlays, there is not enough production capacity. We must significantly improve the development process, increase the production of computer equipment by several times, but alas, the available production capacities in microelectronics itself are not completely suited to providing modern production, which requires special superclean rooms, for example, and it is practically impossible to install these facilities in old buildings. And you know very well how new plant construction goes in our country...

But we must not give up. We must formulate a policy, fix realistic goals, and of course, work hard. After all, it is possible to propose enticing but unsound plans. This has happened on more than one occasion. And thus, we must weigh everything against the realistic possibilities. We are proceeding on the assumption that, as a number of critical problems in the country are solved, it will become possible to build up forces in the area of informatization. A suitable concept has now been worked out on the basis of a competitive version.

The concept is the pillar of the government's policy. It provides the goal posts, it lays the groundwork for the position of the country both internally and in terms of international cooperation. For example, how may computerization be stimulated? By importing foreign-made computers, using the lure of favorable duties or no duties at all, or by relying solely on Soviet industry, while increasing import duties? And should we use imported accessory parts in place of those lacking in Soviet industry or, again, rely on our own production, so that we do not become dependent in future? Create joint

ventures for production of computer equipment, modern switching stations, communications apparatus, or try to produce these by ourselves? There are several possible views.

[Question] That is, there are still a great many questions for which optimal answers must be found?

[Answer] Absolutely. And the concept that has been prepared addresses these questions.

[Question] Which sectors of the economy, in your opinion, are most in need of informatization?

[Answer] Before replying to this question, let me say that our committee does not produce computer equipment. This is done by the enterprises of the other ministries. We are basically concerned with coordinating the nation's efforts in the field of informatization—here as well, we have our share of difficulties. I consider the supply of information to the population a very important field. The showing here is poor—there are almost no home computers, no good software. Of course, we are adapting a lot, and in this area complicated legal issues arise. After all, programming is a highly specific and important field, into which the true value of computer equipment is being shifted. Unfortunately, demand has not yet taken shape in our country and we have no possibility of developing good programs for the population. In order to create these, one requires the tools, and this, in turn, requires hardware of a higher sophistication than that of the average user.

Or take the process of computerization of the schools. We are poor in this regard, the government quotas are not completely filled, we do not always understand the full importance of this problem. There are those—including people in the education system—who think that computerization in the school is done only to give the children an idea of what a computer is, to know its layout and the basic of programming. This is not quite right. The pupil after graduating from school or completing a course in computers should acquire a new ability, he should realize that he cannot work, make decisions, or live without a computer.

I know of no sector where informatization would not produce a positive impact. But, given our limited resources today, I am firmly convinced that informatization of the schools should take place at the expense of any other projects. Above all in the schools. Even ahead of the technical colleges! The foundation has been laid. Computer classes and clubs have been started. Educated, sensible people are in charge of them. We must now work so that everyone understands and accepts this. There is also the problem of teacher training. And once the equipment has been brought to the school and made to work, it should be made to work with maximum efficiency.

At the first collegium of the committee that I attended, we in fact discussed the issues of computerizing the schools. The situation here is complex, there are very few good school computers. Several steps were outlined. It is necessary to introduce new, improved methods for evaluating the equipment, elements of an independent appraisal, to develop for the enterprises supplying school computers a system of taxes and benefits encouraging quality, dependability, and superiority of the particular types of school computerization equipment which they are creating. The independence of the enterprises is taking shape. That they are creating ever newer computer models is good, but the models themselves vary, and therefore there should be an objective assessment, not by the executives, but by special committees, where scientists, interested parties, users, and specialists from industry should evaluate the different computer models and offer benefits to the manufacturers of the best ones, for example, favorable taxes. And higher taxes should be placed on lower-quality computers. It is necessary to work out and boldly implement new economic methods of encouraging computer quality.

[Question] As you know, COCOM has removed the restriction on supply of a number of personal computer models to our country. What are the chances for filling some of our market with imported computers?

[Answer] If we were rich enough and did not have problems with foreign exchange, I would think it necessary to purchase a sufficient quantity of computers and give them to the schools. I am an advocate of purchases, but in the conditions prevailing in our country today the priority in purchasing must go to consumer goods. Therefore, in my view, it is advisable to form joint ventures. This is the policy we support.

Annotations From AVTOMETRIYA, No 2, 1990

907G0159A Novosibirsk AVTOMETRIYA inRussianNo 2, Mar-Apr 90 pp 105-109

[Annotations]

[Text]

UDC 681.786.23/24.014.3

P.I. Goskov, S.P. Pronin, "The Effect of the Size of the Aperture Diaphragm of a Photodetector on the Recording of Fraunhofer Diffraction," AVTOMETRIYA, No 2, 1990.

The effect of the size of the aperture diaphragm of a photodetector (ADFP) on the process of recording Fraunhofer diffraction from a slit is studied theoretically and experimentally and recommendations on the use of methods of processing a distorted valid signal are given. In the theoretical part the derivation of formulas for the calculation of the location of the extrema (minimums and and maximums) subject to the size of the aperture diaphragm of a photodetector is cited. Using the expansion of special functions, a simplified formula of the description of the output signal is obtained. In the experimental part the measurement of the distance

between minimums by a CCD photodetector is analyzed. Tables—1, illustrations—4, references—10.

UDC 621.373.826:621.391.62

V.I. Anikin (deceased), D.A. Letov, S.V. Shokol, "On Several Features of Measurements of the Distribution of Light Intensity at the Focal Point of a Waveguide Lens," AVTOMETRIYA, No 2, 1990.

The features of the functioning of the elements of a circuit for the measurement of the distribution of light intensity at the focal point of a waveguide lens are analyzed. The circuit with the transformation of a surface wave into a body wave on a prismatic coupling element is universal and can be used with any types of waveguide lenses for nondestructive testing when developing the technology of their production. Illustrations—3, references—9.

UDC 621.382:537.22

G.M. Gershteyn (deceased), V.L. Grishchenko, I.A. Matveyeva, L.S. Sotov, "The Restoration of a Latent Electrostatic Image by the Method of Electrostatic Induction," AVTOMETRIYA, No 2, 1990.

For the purpose of reducing information losses when reading an electrostatically charged or charge image (a latent electrostatic image) by electrostatic probing systems the synthesis of probing systems according to frequency responses, as well as the processing of the results of reading by the solution of an integral equation by regularization methods are proposed. Illustrations—6, references—8.

UDC 535.8

D.A. Bezuglov, "The Synthesis of the Wavefront of the Object Field According to the Results of Measurements of a Hartmann-Type Sensor by the Method of Spline Functions," AVTOMETRIYA, No 2, 1990.

An algorithm of the synthesis of the wavefront according to the results of measurements of a Hartmann-type sensor by the method of spline functions, which makes it possible to reduce computer expenditures substantially, is proposed. Expressions which make it possible to switch from the spline representation of the wavefront to the Cerniquet orthogonal basis are cited. A comparative analysis of computer expenditures is given. Tables—1, illustrations—1, references—5.

UDC 681.32(075)

L.S. Korolyuk, B.P. Rusyn, "The Formation of a Set of Informative Attributes on the Basis of the Analysis of Primitives," AVTOMETRIYA, No 2, 1990.

The approach to the formation of a set of informative attributes by the formation of compact invariant descriptions of primitives at the stage of their entry into the recognizing system—at the moment of the scanning of the image by a nonlinear sweep—is examined. It is shown that the obtained descriptions are invariant with respect to affine transformations of primitives and meet the demands that are made on informative attributes. Illustrations—4, references 6.

UDC 621.391

Ye.P. Nechayev, A.P. Trifonov, "The Estimate of the Area of Optical Images Given an Unknown Signal Strength and an Unknown Noise Level," AVTOMETRIYA, No 2, 1990.

A maximum likelihood receiver, which carries out the estimation of the area of an optical image when the strength of the valid signal and the noise level are unknown, is synthesized. The characteristics of the receiver are found. It is shown that it provides a noticeable gain in the accuracy of the estimate as compared with quasi-optimal algorithms. Illustrations—3, references—6.

UDC 519.68

V.A. Gorokhovatskiy, V.I. Strelchenko, "Filters for the Extraction of the Features of Objects in an Image," AVTOMETRIYA, No 2, 1990.

A filter for the extraction of unidimensional fragments of objects, which are characterized by first-derivative jumps at the boundary points, is proposed and substantiated. It is immune to fluctuation noises and is adjusted for the extraction of fragments with durations which lie within a given interval, and with a contrast that is not less than the given contrast. Estimates of the output signal-to-noise ratio are obtained. The results of simulation on a computer are cited. References—4.

UDC 681.3

P.A. Chochia, "A Parallel Algorithm of the Calculation of a Sliding Histogram," AVTOMETRIYA, No 2, 1990.

A parallel algorithm of the calculation of a histogram and ordinal statistics with respect to a sliding fragment, which are used when processing unidimensional or two-dimensional digital signals, is proposed. The algorithm makes it possible to perform calculations for a finite number of actions regardless of the size of the fragment. The block diagram of a multiprocessor that implements this algorithm is described. Illustrations—2, references—11.

UDC 681.325:550.837

M.N. Bukharov, A.N. Vystavkin, V.V. Kondratyev, V.A. Kuznetsov, A.D. Morenkov, A.Ya. Oleynikov, A.N. Revtov, N.A. Tikhomirov, A.V. Elbakidze, "A System of the Management and Acquisition of Data for Work Within a Geophysical Electric Prospecting Complex

Based on a Magnetohydrodynamic Generator," AVTOMETRIYA, No 2, 1990.

An experimental model of an automated system, which is intended for work within a geophysical complex in case of the electric prospecting for petroleum and gas by the method of the formation of a field with the use of a powerful pulsed-action magnetohydrodynamic generator, is developed. The system facilitates prediction of the level of the natural electromagnetic field of earth, synchronization of remote electric prospecting stations, start-up of the magnetohydrodynamic generator, and acquisition, recording, and graphic representation of data. The system was realized on the basis of an Elektronika 60 microcomputer and CAMAC hardware. An interface based on fiber optic systems like the Elektronika MS 4101 and Elektronika MS 8201 was developed for connection to the magnetohydrodynamic generator. Illustrations—2, references—7.

UDC 681.3

N.A. Kutsevich, A.Ya. Oleynikov, "Base Software for Systems of the Automation of the Experiment Based on an Instrument Interface," AVTOMETRIYA, No 2, 1990.

The implementation of the concept of base software (BPS) for systems of the automation of the experiment (SAE) based on an instrument interface and CAMAC hardware is examined. The distributive and the procedure of the generation of base software, which make it possible to implement it for specific applications, are described. A comparative analysis of the labor intensiveness of the programming of SAE systems in accordance with the metrics of M.H. Halsted with the use of base software and a package of intermediate-level subroutines without additional software is made. It is shown that the use of base software is the least labor-intensive. Tables—2, illustrations—1, references—6.

UDC 621.315.592.593.231

S.V. Ivanov, P.S. Kopyev, B.Ya. Meltser, A.N. Rayev, A.P. Solonitsyna, S.V. Shaposhnikov, "An Automated Control System of the Technological Process of Molecular Beam Epitaxy," AVTOMETRIYA, No 2, 1990.

An automated control system of the technological process of molecular beam epitaxy is described. The capabilities of the proposed hardware and software from the standpoint of the reliability and universality of the system are analyzed. A text file with a description of the epitaxial structure in a specialized language, which is close to professional language, is the input information for the system. An example of a structure developed by means of the described system, which reveals its capabilities, is presented. Illustrations—3, references—9.

UDC 528.022.2:621.383.4

N.L. Tsesyul, "A Polarization Photoelectric Converter of Angles of Twist," AVTOMETRIYA, No 2, 1990.

A photoelectric converter of angles of twist using the phenomenon of the polarization of light is described. A mathematical model of the converter and a calculation of the conversion error are given. The theoretical conclusions are confirmed by experimental results. Illustrations—1, references—3.

UDC 681.325:681.787

V.A. Aleshin, "An Automated Measuring Complex Based on Geophysical Laser Interferometer-Deformographs and a Microcomputer," AVTOMETRIYA, No 2, 1990.

The structural diagram and algorithms of the operation of an automated geophysical complex based on two laser deformographs, a microcomputer, and CAMAC hardware are described. The technical characteristics of the complex and the results of calibration are presented. The possibility of using the complex for the study of the spectral-time characteristics of microseismic signals in the frequency band of 0.1-150 hertz is shown. Illustrations—2, references—12.

UDC 681.3+681.883.41

V.I. Kayevitser, A.Ya. Oleynikov, V.P. Sinilo, A.V. Sknarya, V.A. Shubin, "A System for Digital Processing of Sonar Information Based on the TsLANP-0280 Complex," AVTOMETRIYA, No 2, 1990.

A description of the expedition system for digital processing of the signals of a side-scanning sonar system with the linear frequency modulation of the sounding signal is given. This system was constructed using the TsLANP-0280 complex. The results of the work of the system on the acoustic mapping of the ocean bed, on the basis of the example of which its capabilities are illustrated, are presented. Illustrations—4, references—3.

UDC 535.41.088

V.I. Guzhov, B.S. Kotarskiy, "The Effect of Capacity When Quantizing Intensity on the Error on the Determination of the Phase in Systems with Controlled Phase Shift," AVTOMETRIYA, No 2, 1990.

The effect of the number of levels in case of the quantization of the intensity of interference patterns on the error on the determination of the phase difference by the method of the controlled phase shift is studied. Illustrations—2, references—2.

UDC 531.787:535.411

V.P. Kulesh, "The Boundaries of the Dynamic Range of the Interference Meter of Gas Pressure," AVTOMETRIYA, No 2, 1990.

One of the basic factors that limit the dynamic range of the interference meter of gas pressure—the phenomenon of the "gas lens," which is due to the occurrence of a parabolic profile of the temperature of the gas in the cross-section of the optical cell in case of a rapid pressure change—is examined theoretically. The forming optical inhomogeneity defocuses the light beam passing through it and upsets the spatial matching of the light waves that interfere in the photodetector, which can lead to an error in the counting of a large number of cycles of the change of phase. It is established that the allowable rate of change of the pressure, in case of which errors do not occur, decreases with the increase of the current pressure. For the increase of the dynamic range it is recommended to increase the temperature of the gas and to reduce the length and diameter of the optical cell and the diameter of the pupil of the photodetector. Illustrations—6, references—3.

UDC 621.8.087:621.386.1

Ye.N. Vladimirov, T.N. Lavrenyuk, "An Automated Method for the Estimation of the Metrological Characteristics of a Pulse-Height Discriminator," AVTOMETRIYA, No 2, 1990.

A description of the method is presented, estimates for the determination of the metrological characteristics of a pulse-height discriminator according to a random sample from the uniform distribution of the heights of pulses are found, and their unbiased nature and consistency are proven. The hardware implementation of the method is described, and the metrological validity of the proposed method is proven. Illustrations—1, references—5.

UDc 681.325

S.V. Dotsenko, V.Yu. Karlusov, "The Synthesis of Algorithms of Digital Filtering with Allowance for the Statistical Properties of the Process Being Developed," AVTOMETRIYA, No 2, 1990.

The problem of the synthesis of an algorithm of a nonrecursive digital filter with allowance made for the statistical properties of the process being developed and the time or frequency responses of the given filter-prototype is examined. The given filter-prototype may be physically impracticable. Formulas for the calculation of the coefficients of the filter and the error of the processing of information are given. An example of the synthesis of a low-frequency digital filter for a process, which has an exponential correlation function, and a filter-prototype with a square-wave amplitude-frequency and linear phase-frequency response is given, and an interpretation of the obtained results is made. References—3.

UDC 535.4:778.38

B. Dubik, M. Zayonts, Ye. Novak, "A Focusing Phase-Only Synthetic Hologram Mirror," AVTOMETRIYA, No 2, 1990.

A method of calculating the topology of the zones of a phase-only synthetic hologram focusing mirror for the arbitrary relative positioning of the source and its image is proposed. Illustrations—2, references—5.

UDC 621.391

A.V. Savich, Ya.A. Fomin, "The Recognition of Multidimensional Normal Ensembles in Real Time," AVTOMETRIYA, No 2, 1990.

An adaptive procedure of the recognition of multidimensional normal ensembles in real time is developed. Precise expressions which describe the reliability of recognition are given. The task of the optimization of the recognizing system, which reduces to the determination of the minimum dimensions of the attribute space, in case of which the required reliability of recognition is ensured, is posed and accomplished. Illustrations—1, references—6.

UDC 620.179.15

F.M. Zavyalkin, V.A. Udod, "The Two-Aperture Coding of Projections," AVTOMETRIYA, No 2, 1990.

The analysis of the advantages of the use of the coded formation of projections is one of the directions of the improvement of the means and methods of computer tomography. The potential of increasing the resolution in projections by their two-aperture coding is examined. It is shown that the use of apertures of different dimensions (in the plane of the monitored layer) under certain conditions makes it possible to obtain projections with a higher resolution. Illustrations—1, references—3.

UDC 621.396:512

V.A. Gorokhovatskiy, O.V. Sytnik, "Combined Algorithms of the Statistical Estimation of the Parameters of Objects in an Image." AVTOMETRIYA, No 2, 1990.

For the evaluation of the status of objects on an image it is proposed to use recurrent two-stage procedures. First the estimates of the linear functionals from the source image are formed, then the parameters of the objects are calculated as nonlinear functions of these estimates. The high speed and the quite good interference immunity of the proposed algorithms are substantiated. The results of experiments are given. References—4.

UDC 621.378.9:535.82

D.T. Alimov, A.M. Bakiyev, Yu.Yu. Polyak, Yu.I. Sitnik, "The Laser Interference Microscopy of Semiconductor Film Structures," AVTOMETRIYA, No 2, 1990.

The possibility of using active optical systems with an image intensifier for the development of methods of checking and diagnosis of the optical quality of semiconductor film structures, which have been applied to mirror surfaces by means of optical contact, is shown. Single-crystal films of GaSe semiconductor with a thickness of 1-2 micrometers, which are in optical contact with reflecting surface (III) of a Si crystal, were studied. The resolution of this method of determining the value of the disturbance of optical contact given a wave length of lambda equals 0.51 micrometer of the laser being used comes to not more than 0.13 micrometer. Illustrations—1, references—4.

UDC 519.24

Ya.A. Bedrov, "On the Averaging of Experimental Curves with Allowance for the Change of Scales and Displacements," AVTOMETRIYA, No 2, 1990.

The problem of averaging experimental curves with allowance made for their deformations due to changes of the scales of the argument and values, as well as displacements along the axis of the argument, is examined. The statement of the problem for two special cases is given and methods of solving it are proposed. In the first case the scales of the values of the curves and their argument are the variables, in the second case the scales of the argument and the displacements are the variables. The computational aspect of the methods reduces to the use of numerical methods of linear algebra. The effectiveness is shown on a model example.

The "Sura" Computers: New Generations

907G0153A Moscow IZVESTIYA in Russian
11 May 1990 Morning (signed to press 11 May 90) p 1

[Text] The Pensa Production Assocation for Electronic Computing Equipment placed on the market a consumer personal computer, the Sura PK 8000.05. The new compact computer, in contrast to its predecessors, can be ordered with programs in several languages. The Sura 8000.05 is a multi-purpose computer which will be useful in production, in scientific research, and in medical institutions.

The first set of computers is being sent to schools in the Pensa oblast to be used in educational classes.

The association's designers are also preparing for production of an improved computer, the "Sura-3." This 16-bit computer will be compatible with the current computers of leading foreign firms.

In all, the plant produces more than twenty products for the consumer and the economy, from door locks to children's toys.

UDC 518.5

Software for Modeling Physical and Mechanical Fields

907G0070A Kiev KIBERNETIKA in Russian No 6, Nov-Dec 89, pp 62-63

[Article by A. N. Shevchenko]

[Text] The POLYe family of problem-oriented languages and systems [1, 2] was designed to solve boundary-value problems of mathematical physics that model temperature, force, strain, electromagnetic and other types of fields, including interacting physical and mechanical fields. Many applied problems associated with the strength, Q, reliability, stability, service life, etc., calculations of various structures reduce to the solution of such problems.

The POLYe system is a complex set of programming and language facilities aimed at the solution of boundary-value problems for partial differential equations without constraints on the nature of the boundary conditions and the shape of the region and the sections of its boundaries. The system is based on the ideas of the direct methods and theory of R-functions [1], the constructive facilities of which enable one to consider boundary and geometrical information at the analytical level.

The applications of this system comprise scientific research, engineering and education in which the entire spectrum of modern computers are used (from PCs to super computers).

In contrast to application packages and systems based on net, finite-element and boundary-element methods, the POLYe system utilizes the unique capabilities offered by R-function theory for the construction of approximate solutions of boundary value problems while considering the complex geometrical shapes of the regions. The geometric and physical parameters can appear in literal form in the equations of the boundaries of the regions and their sections, and in the structure of the boundary-value problems' solution. This provides the additional functional capabilities of the system when solving optimization problems.

The knowledge base of the POLYe system includes, in addition to the standard algorithmic support, constructive facilities of approximation theory, methods of R-function theory to construct the solution structures, a procession algebra for the exact differentiation of the superpositions of elementary functions and other tools of applied mathematics. The knowledge base possesses the form of a semantic network that implements the model of the object domain and contains, on the one hand, algorithms and application packages, and on the other, sets of various computational schemes, model descriptions, methods and diverse physical and geometrical characteristics. These data express the meaning content of the solution steps of the formulated problems and are treated as knowledge regarding the object domain which is used to execute the required calculations.

The POLYe system traditionally supports the user's interaction with the computer (interpretation of interactive schemes, control of scenarios and display windows, etc.), as well as operation of a knowledge base and application packages.

Information and directive frames are used as the principle knowledge representation elements in the POLYe system. The information frame includes elements to describe the scenario of the interaction and possesses a name which is shown on the display screen in the form of a question. The values of the frame are the possible answer variants. Each frame pertains to a particular solution step of the boundary-value problem. During the dialog the user specifies or resolves the frame by inputting the number of the answer.

The directive frame contains program fragments (macrodefinitions) in the RL language which describe various solution steps of the boundary-value problem. Adequately complete sets of such macrodefinitions are included in the library of the POLYe system and together determine the program packages for the solution of particular boundary-value problem classes in mathematical physics.

Knowledge in the POLYe system is stored in the form of a semantic network which is used to generate a hierarchy of macrodefinitions in the knowledge base and is used to control the coarse of the dialog. This network is represented by a set of rules in the form of productions IF <condition> THEN <action>, where <condition> is a logical function that unites the names of the information frames, and <action> includes primarily the frame parameters' acquisition of values.

The primary structure of knowledge in the POLYe system possesses three clearly expressed levels: applied, mathematical and software. The information and directive frames designed to construct algorithms and programs when solving the boundary-value problem are located at the lowest level. It is here that most active dialog with the system occurs, and it is assumed that the user possesses adequately deep knowledge regarding the object domain (models, methods, algorithms and programs).

The interactive interaction regarding the mathematical formulation of the problem occurs at the middle level. Here many frames of the bottom level are automatically resolved.

At the top level the formulation of the problem is accomplished in terms of engineering concepts associated with the calculation of physical and mechanical fields. The choice of a mathematical model, problem solution algorithm and corresponding program is accomplished practically without the user's intervention. In exceptional cases the POLYe system may present the

user with a leading or specifying question regarding the formulation of the problem and the form of presenting the obtained results.

The POLYe system automates (using the proposed new methods) the programming and solution of boundary-value problems of mathematical physics and includes hardware, algorithmic, software, informational, linguistic and methodological facilities.

The hardware of the POLYe system includes such computers as the BESM-6, the YeS, the SM and the IBM PC with standard peripheral equipment set.

The algorithmic support of the POLYe system is provided by the aggregate of following methods: variational and projective methods of solving boundary-value problems for partial differential equations; and the method and constructive facilities of R-function theory to transform geometric information into analytical information and construct the solution structures of boundary-value problems.

The software of the POLYe system is represented by a wide variety of program packages: differentiation of the superposition of complex functions; integration of functions in regions of complex shape; implementation of the constructive facilities of R-function theory; solution of linear algebra problems; and geometric information input and output.

The informational support of the POLYe system includes specialized libraries that are characterized by a large diversity of stored information, ranging from the results of computational experiments to source text of programs.

The linguistic support contains: a language superstructure above a procedurally-oriented programming language (extension of FORTRAN-77); the problem-oriented language RL to describe the boundary-value problems of mathematical physics; and nonprocedural languages to describe the problem utilizing the terminology of a particular field of knowledge.

The input languages of the POLYe system provide convenient user interaction with the computer system and arrange the process calculations of the physical and mechanical fields.

The methodological support includes a description of the application, a description of the program, a description of the language and a programmer's manual.

The user's work in the POLYe system when problem solving includes making scientific investigations and engineering calculations, and is broken into a number of steps: investigation of physical entities and effects, the construction of mathematical models and the formulation of the problem; creation or selection of efficient methods of solving the problem, and developing an algorithm; composition of a program that implements this algorithm in the POLYe system; construction of control texts, data preparation, program debugging; making computer calculations; processing and output of the calculation results; and analysis of results and reaching decisions and conclusions.

A sophisticated programming environment is provided by the automated synthesis of the working programs; for example, utilizing the POLYe system, a computer experiment can be conducted in several hours or days (depending on the complexity of the problem, the user's experience and the capabilities of the system). Many months are sometimes required to solve the same problem using standard programming tools. Here significant difficulties of both a technological and psychological nature frequently arise which are associated with the large volume of processed information, the complexity of the computational algorithms, the diversity of mathematical experiments and the many models of the formulated problems.

BIBLIOGRAPHY

1. Rvachev, V. L. "Teoriya R-funktsiy i nekotoryye yeye prilozheniya," [Theory of R-Functions and Several Applications]. Kiev: Nauk. dumka, 1982, 550 pp.

2. Rvachev, V. L., Shevchenko, A. N. "Problemno-orientirovannyye yazyki i sistemy dlya inzhenernykh raschetov," [Problem-Oriented Languages and Systems for Engineering Calculations]. Kiev: Tekhnika, 1988, 198 pp.

UDC 51:681.3.06

Structural Design and Synthesis of Certain Symbol-Processing Programs

907G0070B Kiev KIBERNETIKA in Russian No 6, Nov-Dec 89 pp 78-79

[Article by A. L. Nagornaya]

[Text] One of the most important problems in modern programming technology is the creation of facilities to generate (synthesize) programs according to their formal specifications. One promising direction in solving this problem is associated with the development of structured programming, whose application promotes a significant reduction of development time, increased reliability, ease of program maintenance, and facilitation of their modification when adapted to the surrounding computing environment, etc.

The application of structured programming tools and methods acquires particular relevance in connection with the organization of parallel calculations on multiprocessor computing systems.

The apparatus of algorithmic algebra systems [SAA], which is in complete correspondence with the structured programming concept, may be used as a technology to construct programs, while the introduction into the signature of the algorithmic algebras of facilities to formalize parallel calculations enables one to view this

theory as a framework for structured programming [1]. The apparatus of SAA and their modifications are oriented to the formalized representation of structured schemes of sequential and parallel algorithms and programs, including those oriented to symbolic multiprocessing (parallel syntactical analysis, translation, etc.).

This apparatus was used to prototype the PKOBOL system for parallel translation from COBOL for a multiprocessor computer complex.

The PKOBOL system is a syntax-driven translator. The analyzer of the system is oriented to the class of context-free languages formalized in terms of t-grammars [2] that provide no-deadlock top down syntactical analysis.

In order to increase the error localization effectiveness, a multi-level representation of the input language grammar is proposed. The essence of this representation consists in the following.

Let G^0 be a t-grammar of the input language

$$G^0 = (\Sigma^0, V_n^0, \sigma^0, P^0),$$

where Σ^0 is the terminal and V^0_n the nonterminal alphabets, σ^0 an element of V^0_n is an axiom of the grammar, and P^0 is the set of rules of the given grammar. The rules of the grammar G^0 consist of one or several disjunctive terms, each of which begins with a terminal symbol:

$$y_i ::= a_{i_1}\alpha_{i_1} \mid a_{i_2}\alpha_{i_2} \mid \dots \mid a_{i_{k_i}}\alpha_{i_{k_i}},$$

where the a_{ij} ($j = 1, 2, .., k_i$) are different terminal symbols. For a level-by-level representation of the grammar G^0, we fix a certain set of terminal symbols in it $a_1, a_2, ..., a_k$ elements of Σ^0. To each of the symbols a_i we juxtapose a certain subgrammar

$$G^i = (\Sigma^i, V^i_n, \sigma^i, P^i),$$

where the intersection of the V_n and V^i_n is equal to the null set, and the rules P^i possess a structure similar to the structure of P^0 ($i = 1, 2, ..., k$). As a result we obtain a two-level grammar of the form $G = (\Sigma, V_n, \sigma_n, P)$, where

$$\Sigma = \Sigma \setminus \{a_i / i = 1, 2, \dots, k\} \cup \bigcup_{i=1}^{k} \Sigma^i; V_n = V_n^0 \cup \bigcup_{i=1}^{k} V_n^i; \sigma_r = \sigma^0:$$

$$P = P^0 \cup \bigcup_{i=1}^{k} P^i.$$

The process of the level-by-level partitioning of the grammar G can in turn be continued.

The level-by-level representation of the input language grammar can be made both from the top down and the bottom up.

This representation of the input language grammar of the PKOBOL system made it possible to reduce the amount of memory occupied by the grammar and accomplish the level-by-level debugging of the grammar while introducing corrections only into the corresponding subgrammars.

The PKOBOL system analyzer was designed by the multilevel structured design method and was synthesized by means of its "Multiprotsessist" tools [3].

"Multiprotsessist" is a system for the structured synthesis (generation) of programs and algorithms according to their formal specifications. The input for this system is the SAA/1 algorithm scheme construction language, which is oriented to the multilevel design of algorithms and programs (both sequential and parallel). The automatic synthesis of the programs in one of the target languages connected to the system is accomplished according to the SAA scheme, the programs' design as formulated in the SAA/1 language and the elementary module library which determines the selected object domain.

We shall list certain features of the system: universality of purpose; the capability of parametrically adjusting to various object domains; utilization of a well-developed algebraic grammar apparatus oriented to the formalization of parallel algorithm classes; multilevel design of algorithms and programs in a language close to natural language; portability of the generated programs; and good documentation.

The analyzer algorithm was designed in the SAA/1 language, which is the input language of the Multiprotsessist system. A PL/1 program for the operating system of the YeS computer was obtained and debugged with its aid. The library of elementary modules was written in PL/1.

Let us present a description of the analyzer in SAA/1 and the notational conventions.

* will denote the end marker of the analyzed line L, while M_1 and M_2 are stacks.

M_1 contains the axioms of the subgrammars which are invoked for analysis. Initially at the top of M_1 is contained an axiom of the grammar σ_n; when separating from the input line the symbols $a_1, a_3, ..., a_k$, which are reserved terminal symbols, the axioms of the corresponding subgrammars are placed in M_1. Upon termination of the syntactical decomposition according to the subgrammar, its axiom is removed from the top of the stack and a jump is made to the subgrammar whose axiom appears at the top of M_1.

The M_2 stack contains the "remainder" from dividing the rule by the terminal symbol.

M_e is the set of nonterminal symbols that contain the empty word e.

S is an index which moves over the rules of the grammar.

S = 1 if the index is located within a rule.

S = 0 if the index is located in front of the first symbol of a rule.

```
ANALYZER SCHEME ====
   "Syntactical analyzer of the PKOBOL translator.
   Input language COBOL; output language PL/1"
=========================== "ANAL"===
  WHILE 'SIM NOT EQUAL TO *'
    REPEAT "SEPARATE SYMBOL FROM INPUT
    LINE"*
    "DEL"* "FORM OUTPUT LINE"
  END
END "ANAL"
"DEL"===
  CASE
  (S = 0 -> "SEARCH FOR APPLIED PRODUC-
  TION"; S = 1 -> "WORK WITH M2")
END
=====
"WORK WITH M2"====
  IF THE SYMBOL AT THE TOP OF M2 = TER-
  MINAL
  THEN
    IF 'SYMBOL = SIM'
    THEN
    ("S = 1"* "SHIFT CONTENTS OF M2 UP BY
    ONE SYMBOL")
    ELSE "SOSH" [not further identified])
  ELSE ("S = 1" "SEARCH FOR APPLIED PRODUC-
  TION");
=====
"SEARCH FOR APPLIED PRODUCTION"===
  "SELECT PRODUCTION FROM TOP OF M1"* IF
  'START OF PRODUCTION = NONTERMINAL'
    THEN
    "SOSH"
    ELSE
    IF 'START OF PRODUCTION = SIM'
    THEN
    "REWRITE REMAINDER OF PRODUCTION
    DIVISION TO M2"* "SET S = 1)
    ELSE
    "SEARCH FOR NEW PRODUCTION"
"SEARCH FOR NEW PRODUCTION"===
  IF 'NUMBER OF DISJUNCTIVE TERMS OF PRO-
  DUCTION IN M1 = 1'
  THEN
  "CHECK Me"
  ELSE
  WHILE 'NUMBER OF DISJUNCTIVE TERMS OF
  PRODUCTION M1 NOT EQUAL TO 0'
  REPEAT
  "JUMP TO NEXT DISJUNCTIVE TERM"*
  IF 'SYMBOL = SIM' THEN "REWRITE
  REMAINDER OF PRODUCTION TO M1"* "SET S
  = 1"
END
```

The analyzer algorithm is obtained through further specification of the meaning operators and conditions in the equalities.

As a result of the syntactical analyzer's operation, the syntactical structure of the input program is not only established, but the calling sequence of the semantic procedures of the object code generator is fixed. The expression of the tie between the input language semantics by means of semantic procedures was found to be a convenient tool for developing the new version of the analyzer.

The experience of working with the "Multiprotsessist" system when prototyping the syntactical analyzer for the YeS computer confirmed the fact that the software product created by the tool system is portable, i.e., it features flexible adaptability to various computing systems. In fact, an analysis of the ASS/1 schemes of the algorithm showed that when transferring the algorithm on a minicomputer to the new operating environment, one must describe the data and develop a library of elementary modules in the chosen target language for the meaning operators and conditions. The scheme of the algorithm, which has been developed and debugged, remains unchanged.

The "Multiprotsessist" system enables one to accomplish multilevel design in accordance with the multilevel structured design method through formalization of the designed entities and the creation of a multilevel system of abstract modules. Here the model is represented as interacting control and operation substructures at each level of abstraction. The control structures are used in the given algorithm to model the separation, inspection and comparison of the input data and the grammar rule symbols, etc. The control structure serves as a model of the processing strategy.

The method of organizing the input data, language grammar, etc., is abstracted here at the algorithm construction level. These problems are solved upon creation of the elementary module library. This feature of the "Multiprotsessist" system also made it possible to avoid altering the scheme of the algorithm when transferring it to a new operating environment.

The descriptions of the algorithm in the ASS/1 language simultaneously serve as documentation for the generated program. The input language of the system is close to natural and permits comments which are convenient for documentation; modification of the program is accomplished by introducing the changes into its SAA scheme with subsequent generation of the new version of the program.

BIBLIOGRAPHY

1. Glushkov, V. M., Tseytlin, G. Ye., Yushchenko, Ye. L. "Algebra. Yazyki. Programmirovaniye," [Algebra. Languages. Programming]. Third edition, revised and expanded. Kiev: Nauk. dumka, 1989, 376 pp.

2. Nagornaya, L. I., Tseytlin, G. Ye., Yushchenko, Ye. L. "On One Method of Multilayer Syntactical Analysis for Parallel Programming Systems," KIBERNETIKA, 1979, No 1, pp 90-97.

3. Yushchenko, Ye. L., Tseytlin, G. Ye., Gritsay, V. P., et al. "Mnogourovnevoye strukturnoye proyektirovaniye program. Teoreticheskiye osnovy, instrumentariy," [Multilevel Structured Design of Programs. Theoretical Bases and Tools]. Moscow: Finansy i statistika, 1989, 208 p.

UDC 681.3.06

Tools to Synthesize Programs for Mini- and Microcomputers

907G0070C Kiev KIBERNETIKA in Russian No 6, Nov-Dec 89 pp 90-92, 94

[Article by V. P. Zakhariya and L. M. Zakhariya]

[Text] The necessity of accelerating the design, development and operation of the software of computing systems requires the creation of highly efficient programming technologies. The multilevel structured design method enables one to create a set of programming facilities aimed at solving this problem [1]. This method is a further development of the structured programming method, in accordance with which the logic of any program is represented in the form of a superposition of a small number of unified logical structures (compositional, distributive, cyclic) [2]. It contains facilities to formalize parallel computations (synchronous and asynchronous disjunction of operators, filter, synchronizer, control point) [3], and combines the formal apparatus of algorithmic algebra systems [SAA] with formal models of languages.

The described system comprises the tool support for the multilevel structured design method in the OS RV and RAFOS operating systems for mini- and microcomputers. In terms of its external interface, it is similar to the MULTIPROTSESSIST system [4] created earlier for the YeS computer, and is favorably distinguished from it in terms of implementation, which made it possible to improve its characteristics and enhance its functional capabilities. In this connection, the following characteristic features of the described system will be identified.

—The system is invariant with respect to the two operating systems RAFOS and OS RV, where it can operate in any version of each of them. Since the system was completely written in the MACRO-11 language, this invariance was achieved in certain cases through emulation of the macrocommands of one operating system in the other. In cases associated primarily with the peculiarities of the file systems of the OS RV and RAFOS, separate file processing code fragments were written for each of the operating systems utilizing conditional translation directives.

—Structurally the system consists of a set of logically complete mutually independent components. The number of such components increases as the tool system is developed, and in each specific case, the minimum system configuration is generated for operation that supports the functions needed by the user.

—Most components of the system are, functionally, translators; all of them are described uniformly in terms of translating LL-1 grammars which are interpreted by the same LL-1 analyzer, while in the implementations of the MULTIPROTSESSIST system for the YeS computer and the Neyron personal computer, each component comprises an LL-1 translating grammar processing program, thus duplicating the syntactical analysis algorithm. The creation of a universal LL-1 analyzer that interprets an arbitrary translating LL-1 grammar enables one, when transferring the system to other machines, to implement only the analyzer and the set of semantic procedures, while the remaining components are described in the internal language of the analyzer and are transferred without change. This approach assures good reliability of the components' operation, and the ease of their debugging, maintenance and modification, since all of these processes reduce to working only with LL-1 grammars, enabling one to digress from the translation mechanism.

—Owing to the syntactical structure of the ASS/1 language (the composite operators and conditions are defined as grammar rules) and the presence of a unified mechanism of syntactical analysis and the execution of semantic procedures, the capability of interpreting the logical ASS schemes is provided. This capability is lacking in all other versions of the system.

—The system occupies up to 15 kilobytes of main memory in the RAFOS operating system, and up to 25 kilobytes in the OS RV, which is on the average two or three times less than with the other system versions. This was achieved through the modularity of the system and the specific features of processing the input and output information by each system module (the data is dynamically supplied from/to external media and occupies no more than 512 bytes in memory). This capability is absent in other implementations.

—The system comprises a generator of programs designed by the multilevel structured design method in the form of logical SAA schemes. Currently the system supports the generation of programs in Pascal, FORTRAN, MACRO-11 and Analytic. Programs are generated in any language by means of system-resident macrogeneration facilities according to a library of macrodefinitions (presented by the user) of the principle ASS/1 structures in the user-selected programming language. The mechanism for implementing a new language is so simple that the system user can connect up the programming language required in several days by following the instructions.

—The simplest debugging facilities, in particular, program execution tracing in terms of the elementary operators and conditions of its SAA/1 scheme, can be generated in the synthesized programs by the system upon the user's request.

—Interactive facilities for optimizing identity transformations of the SAA schemes (the OPTIMA system) for SM-1410 computers are connected to the system.

—The system is most effective when developing programs having a complex logical structure and relatively simple data structures, and offers the ability of synthesizing recursive programs according to recursive SAA schemes.

Design of the Software Product in the MULTIPROTSESSIST/SM System

The design process of the software product with the MULTIPROTSESSIST/SM system consists of several steps. At the first step involves the creation of a hierarchically interconnected family of logical SAA schemes that describe the logical structure of the created software product in the form of a superposition of SAA operations over a set of operators and conditions which are elementary with respect to the design's level of detail. The family of SAA schemes is described in the system input language: SAA/1. SAA/1 is a universal algorithmic language for designing the logical schemes of programs, and contains convenient mnemonics for representing the primary SAA operations, and facilities for representing the elementary operators and conditions as well as the principle design levels in the form of meaning identifiers which convey information about their purpose, application conditions, and the nature of the actions produced by them. Thus, the logical SAA scheme in the SAA/1 language represents the algorithm in a form close to natural language, which is extremely important for documenting and further maintenance of the software product. At the next design step involves the selection of the primary programming language in which the design must be implemented, and the independent description in it of the elementary operators and conditions of the logical SAA schemes. The subsequent steps are the synthesis of the programs according to their SAA schemes and basis, the direct debugging of the synthesized programs, and their optimization by the apparatus of SAA identity transformations. The software product finally obtained possesses a clear and unique description in the form of an SAA scheme, which at the maintenance step may fulfil the role of documentation, is optimized according to the defined criteria, and is quite easily corrected. The corrections reduce to either changes in the SAA schemes or to separate modifications of the basis which are independent of the scheme, after which the system automatically generates the new variant of the software product.

The MULTIPROTSESSIST/SM system enables the generation of programs without certain (and even all) basis implementations. In this operation mode, the system generates replacements of the absent operators and conditions. Automatic generation of debugging facilities is also possible upon the user's request.

The text of the SAA scheme and the basis implementation file are fed to the input of the MULTIPROTSESSIST/SM system, while at its output the user obtains the program in the form of two components: control and operation structures. The control structure reflects the SAA scheme in the chosen programming language, while the operation structure comprises a set of procedures and functions that implement the basis of the SAA scheme, the set of its elementary operators and conditions. When specified by the user, the elementary operators and conditions of the basis can be inserted into the text of the control structure without implementation as procedures, which optimizes the execution time of the created program.

The elementary operator and condition implementation file may contain the implementations of the elementary elements of not only a specified SAA scheme, but of the entire object domain. In this case the generation of programs for the given object domain is reduced only to the design of SAA schemes over an elementary basis fixed in this way, and the subsequent software product is created automatically by the system.

Logical Structure of the System

The system consists of a universal analyzer which accomplishes syntactical analysis by any translating LL-1 grammar, a unit for adjusting the analyzer to the particular operating system (OS RV or RAFOS), a set of translating LL-1 grammars that describe the various steps of synthesizing the software product, a universal macrogenerator, and a set of selected utility programs. The analyzer and the unit for adjusting it to the particular operating environment comprise the core of the system. This is the most machine-dependent part of the system, since the problem of transferring the system from computers of one type to another reduces primarily to the implementation of these components. In order to simplify this procedure, the components are structured, and the most machine- dependent parts (processing of the input symbol flow during translation, anomalous situation handling unit) are separated in them, and detailed SAA schemes of their operation are used. The analyzer was implemented as an abstract mechanism that processes a particular set of instructions, where the number and form of representation of the instructions are easily corrected by modification of an instruction table. An arbitrary translating LL-1 grammar is represented as a sequence of analyzer instructions and functionally comprises a program for it that defines the entire translation process. Upon transfer to other types of computer, the grammars' representation does not change. The following are analyzer instructions: to process terminals and nonterminals, initialize the semantic translation procedures, and execute these instructions sequentially, conditionally and repeatedly.

System Operation

The system operation is broken down into several steps. At the first step yields a syntactical analysis and translation of the initial SAA scheme into an algebraic formula, in which the identifiers of the composite and elementary operators and conditions of the scheme are standardized according to type and identity. The formula obtained is a logical skeleton of the scheme that does not carry any information regarding the semantics of the actions produced by it. In this form it is easy to apply identity transformations to it, thus it is expedient to add the OPTIMA transformation subsystem after the first step. A table of correspondence of the meaning identifiers of the SAA scheme to the standard identifiers of its formula is also constructed at this step. The contraction of the formula can be accomplished at the first step upon the user's request. Contraction is an identity transformation in which the formula of the scheme is reduced to a single SAA expression by substituting specifications of all the composite operators and conditions, i.e., it is represented by a single equation. This transformation provides a significant execution time optimization of the program generated in the subsequent steps. In order to generate programs in MACRO-11 and in parallel programming languages, the scheme formula must be developed. Development is an identity transformation of the formula that is the inverse of contraction, after which, due to the introduction of new composite operators and conditions, the formula is transformed into a set of equalities, each of which is determined by a single SAA operation. The subsequent steps accomplish the direct generation of the software product according to the formula of the SAA scheme and the set of implementations of the scheme basis. Here the programs are generated uniformly in different programming languages. The LL-1 grammar of the SAA scheme formula language, which is represented in the form of the analyzer program, constructs a chain of macrocalls according to an arbitrary SAA scheme formula that uniquely define the principle SAA structures in various programming languages. The chosen set of macrodefinitions considers the general and specific features of modern programming languages when interpreting the SAA operations in them. Final generation of the software product is accomplished according to the chain of macrocalls, the file of implementations of the elementary operators and conditions of the scheme in the programming language and the set of macrodefinitions of the primary SAA structures for this language. A given synthesis step is validated by the macrogenerator created for this purpose.

In connection with the narrow range of solved problems, the structure of the macrogenerator is quite simple. It processes only macrocalls with parameters (or without them) without permitting imbedded macrocalls, recursions or more complex structures. In addition, since the macrocall file is automatically generated, the appearance of syntactical errors is eliminated, which in turn eliminates the necessity of processing and diagnosing them.

The macrocall file consists of an arbitrary sequence of symbol lines and macrocalls. Each macrocall is written in a separate line and begins with a special symbol (or any other symbol specified by the user in a response to a prompt), followed by the name of the macrocall and a parameter list.

The macrodefinition file for each language in which the program is generated is a sequential file. No library is needed since the number of macrodefinitions is small, as a rule.

The operation of the described macrogenerator possesses several features.

—If the sought definition is not encountered in the set of macrodefinitions, then nothing is generated in the output file and no diagnostic messages are generated. This is extremely useful since it enables one, after having created an extra macrocall file, and varying only the set of macrodefinitions, to generate programs in different programming languages utilizing the same macrocall file. The extra macrocalls are necessary since almost every programming language possesses constructs unique only to themselves, for example, the FORWARD procedure in Pascal.

—The macrogenerator does not process all macrocalls identically. There are two groups of macrocalls, and their names are specified in particular tables which are processed specially. The first group includes macrocalls which are processed according to a key which is specified when the macrogenerator is called. This makes it possible to incorporate debugging facilities into the generated program. Another group consists of macrocalls whose parameters are the meaning identifiers of the SAA scheme. They are replaced by implementations of these entities in the programming language if such implementations exist; otherwise, a model of the procedure or function corresponding to this entity is created.

Conclusion

The system was used to create the TEKHNOLOG automated system for production preparation (stamping parts from sheet metal) and a computer-aided design system for optical and mechanical objects. The expediency of its application in implementing methods of computational mathematics was investigated. An automated work station for programmers in homogeneous computing environments is being created on the basis of the system. Currently an attempt is being made to utilize the system to create expert systems.

BIBLIOGRAPHY

1. Glushkov, V. M., Tseytlin, G. Ye., Yushchenko, Ye. L., "Multilevel Structured Program Design. Method Formalization and Applications," KIBERNETIKA, 1981, No 4, pp 42-65.

2. Glushkov, V. M., Tseytlin, G. Ye., Yushchenko, Ye. L. "Algebra. Yazyki, Programmirovaniye," [Algebra. Languages. Programming]. Kiev: Nauk. dumka, 1978, 320 pp.

3. Glushkov, V. M., Tseytlin, G. Ye., Yushchenko, Ye. L. "Metody simvolnoy multiobrabotki," [Symbolic Multiprocessing Methods]. Kiev: Nauk. dumka, 1980, 242 pp.

4. Gritsay, V. P. "On the Implementation of the Structured Programming Tool System of the MULTIPROTSESSIST System," KIBERNETIKA, 1983, No 3, pp 118-123.

5. Petrushenko, A. N., Vayserman, L. M. "Avtomatizatsiya preobrazovaniy program v sisteme MULTIPROTSESSIST: Tez. dokl. 7-y Vsesoyuz. shk.-seminar 'Parallelnoye programmirovaniye i vysokoproizvoditelnyye sistemy'," [Automatization of Program Transformations in the MULTIPROTSESSIST System: Theses of Reports of the Seventh All Union School-Seminar "Parallel Programming and High-Performance Systems"]. Kiev: IK AN UkrSSR, 1986, pp 82-84.

Microprocessor System To Control Pulse Electrolysis

907G0080A Kishinev ELEKTRONNAYA OBRABOTKA MATERIALOV in Russian No 5, Sep-Oct 89 pp 72-73

[Article by N. A. Kostin and V. N. Siromakha]

[Text] One of the promising methods of obtaining high quality functional electrochemical coatings is the application of programmed pulse electrolysis processes [1]. The most important electrochemical factor of pulsed deposition processes is the cathode potential. Controlling the cathode potential during application of the coatings makes it possible to provide them with specified properties.

The programmed sources currently used to supply electroplating tanks enable one to vary the cathode potential by changing one or several parameters of the polarization current pulses. The source programs are written on the basis of dependencies between the current pulse parameters and the properties of the applied coating (conductance, microhardness, etc.), which are experimentally obtained. However, the magnitude of the cathode potential also depends on a number of other factors: composition of the electrolyte, the solution temperature, the condition of the anodes, etc. It is an extremely difficult task to keep this set of factors constant and equal to their values when obtaining the empirical dependencies.

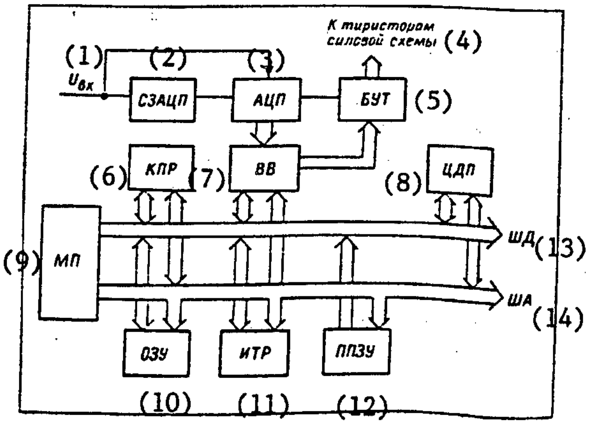
The effectiveness of controlling the properties of the obtained coatings can be increased by introducing a feedback in the power source-electrolyzer system. The properties of the coating cannot be directly controlled during the deposition as a result of significant technological difficulties. Thus an alternative solution is introducing a cathode potential feedback.

The authors have developed a microprocessor system to control pulse electrolysis, which enables one to compare the specified potentials stored in an erasable read- only memory with the measured potentials and issue control actions. It further facilitates the issuance of a control action that determines the cathode potential through corresponding measurements of the amplitude, frequency and relative duration of the power supply pulses when given the cathode potential's time dependence curves.

The system was developed on the basis of the MS 27.02 programmable controller. The MS 27.02 was selected due to its low cost and its functional capabilities, which are fully adequate to the solution of the given problem. The controller incorporates the widely used K580 microprocessor.

The authors consider most expedient the use of the newly developed pulse electrolysis control system as one of the lowest stages of a hierarchical microprocessor system to control the electroplating processes, although autonomous operation is also feasible.

Block Diagram of Pulse Electrolysis Control System



Key:—1. Input voltage—2. Analog to digital converter trigger circuit—3. Analog to digital converter—4. To the thyristors of the power circuit—5. Thyristor control unit—6. System interrupt controller—7. Input- output unit—8. Digital display—9. Microprocessor—10. Random-access memory 11.Interval timer—12. Erasable read-only memory—13. Data bus—14. Address bus

Power supply current pulse frequencies of tens of kilohertz can be used in the pulse electrolysis process. The cathode potential varies at a similar frequency. Effective process control is only possible utilizing the instantaneous values of the cathode potential. This places rigorous temporal constraints on system operation. One way to solve the problem of increasing the system's speed is by analyzing only the extreme values of the cathode potential, all the more since they determine to a significant degree the properties of the obtained coatings. Thus the extreme values should be registered by hardware, since the low speed of the K580 microprocessor used in the system does not make it possible to do this with software.

A block diagram of the pulse electrolysis control system is shown in the diagram.

The system operates as follows. The potential difference between the cathode and comparison electrode arrives at the inputs of the analog to digital converter and the analog to digital converter trigger circuit. The F7077/12 series- produced analog to digital converter is used in the system, whose conversion time is 3 microseconds. The trigger circuit generates pulses that trigger the analog to digital converter when the electrode potential achieves the extreme values. Upon termination of the specified initialization time interval, the interval timer interrupts the execution by the microprocessor of the background program through the system interrupt controller and a jump is made to an interrupt handling routine. The interval timer and system interrupt controller consist of series K580 integrated circuits, the K580 VI53 and K580 VN 59 series, respectively. The interrupt handling routine restarts the interval timer. The measuring results are read via an input-output device fabricated from K580

VV55 integrated circuits. The microprocessor compares the measuring result with the cathode potential which is stored in the memory and corresponds to the current moment of time. If the difference between the measured and specified cathode potentials exceeds the permissible value, a control action is issued (considering the sign of the difference). By "control action" is meant the alteration of the current pulse parameters by a certain fixed amount. The new parameter values enter the thyristor control unit through the input-output unit. The programs that determine the system operation and the file of numbers that determines the potential variation law during the coating deposition process are stored in an erasable read-only memory (K573 RF5 integrated circuits). The capacity of the read-only memory in the system is 4 kilobytes. A random-access memory is also used in the system. The capacity of this memory is 1 kilobyte (K565 RU2A integrated circuits).

The background program indicates process information on a digital display. The microprocessor, input-output unit, system interrupt controller, interval timer, random-access and erasable read-only memory and digital display functional units are part of the reprogrammable MS 27.02 controller. The analog to digital converter trigger circuit and thyristor control unit primarily consist of K155 integrated circuits. Assembly language was used to write the software.

Currently work is being conducted to introduce the newly developed system at an electronics fabrication plant.

BIBLIOGRAPHY

1. Zabludovskiy, V. A., Kostin, N. A. "Obtainment of Microlayered Electrochemical Coatings by Programmable Pulse Electrolysis Processes," ELEKTROKHIMIYA, 1987, 23, No 6, pp 734-739.

UDC 025.4:336.12

Integrated Budgetary Information Classification and Coding System

907G0055A Moscow KLASSIFIKATORY I DOKUMENTY in Russian No 12, Dec 89 pp 20-24

[Article by G. I. Zhukovskiy, Byelorussian Department of the All-Union Scientific Research and Design Institute on Statistical Information of the USSR State Committee on Statistics]

[Text] The significant quantity of accounting information and the interrelationship between data on economic and management activities of budgetary departments and organizations impose requirements on information classification and coding systems in order to obtain analytic information, and to compare performance factors accounting for economic groups and to produce periodic and annual reports.

The specific nature of budgetary accounting does not make it possible to utilize effectively classifiers developed by different ministries and agencies for their departments and enterprises.

The Byelorussian Department of the All-Union Scientific Research and Design Institute on Statistical Information of the State Committee on Statistics has begun to develop a unified accounting nomenclature classification and coding system for departments and organizations which are part of the USSR state budget; this organization has developed a "standard project for the comprehensive automation of accounting operations for centralized accounting departments which are under the USSR state budget using punched card computers."

The standard project includes classifiers that are common to all budgetary departments of public health, education and cultural systems. The classifiers have been coordinated with the Ministry of Finance of the USSR and have been published by the publishing house "Finance and Statistics" (1981). Book 1 describes the information processing routine on punched card computers. Books 2-5 contain the classifiers themselves ("Accounting of Financial Operations, Funds, Cashier and Actual Outlays," "Accounting 01. Primary Assets," "Accounting 06. Materials and Food Supplies," "Accounting 07. Inexpensive and Wear-Prone Objects").

Between 1980 and 1981 these classifiers were tested during the incorporation of comprehensive accounting mechanization in Bobruysk for the centralized accounting departments of the Municipal Department of Public Education, the Municipal Department of Public Health, the Central Emergency Medical Service, the Central Children's Hospital, the Municipal Department of Culture, Municipal Professional and Technical School No. 91; the Bobruysk Regional Educational Department, and the Regional Department of Public Health based on the Mogilev Oblast information computer center (the Byelorussian SSR); the centralized accounting departments of the regional educational departments and the regional department of public health of the city of Kuba in the Azerbaydzhan SSR; at four centralized accounting departments of the Voroshilov and Frunze regional educational departments and regional departments of public health in Moscow based on the departments of the main statistical management computer centers in Moscow. The standard project and unified classifiers were introduced for commercial use and were recommended for uniform adoption. The comprehensive accounting mechanization system began to be widely introduced at the centralized accounting departments of budgetary departments beginning in 1982 using punched card computers.

The standard design and unified classifiers of accounting nomenclatures represent an important condition for the broad utilization of comprehensive accounting mechanization at the centralized accounting departments. After this system was introduced iu(was no longer necessary to carry out one of the most labor intensive and difficult

stages of operations: surveying, collection and systematic analysis of accounting positions (analytic accounting facilities). This was confirmed by the practice of coding primary assets, material supplies, low-value and wear-prone objects. The nomenclature of material goods accounting in medium- and large-scale organizations totals 30-50,000 records with identical names idea to implementation. This is particularly important in the case of small-scale and flexible manufacturing conditions. The idea itself is not the only original aspect: the "packaging" proposed by the designers was also original.

"We incorporate all customer-specified modifications in the design and provide a one year guarantee and at the end of this year we provide an updated system at no charge," says Tyurin. There are many other additional services. In order to break into the international market it is necessary to learn to live by its rules. One of these rules is that a new product must have the newest user qualities.

Yet isn't it impossible to work continuously at such a pace as you have done?

"Today this is not only possible but is in fact necessary," my interviewee notes with some confidence. "This, of course, is true if we want to take an appropriate place among the developed nations. Judging both by my experience and that of my colleagues, specialists are yearning to work at full speed. If all the stored up energy were to be released it is entirely possible that the world would speak of the 'Russian miracle.' For example, even today we are prepared to cooperate with any interested partner both domestically and abroad."

These words are not a bluff. Proof of this can be found in the "Elektronika-2000" International Exhibition which recently was sponsored by the American-Polish company IDM at the USSR Exhibition of National Economic Achievements; this company showed products at the exhibit containing software developed by Tyurin's team. And this was the only exhibit of domestic, Soviet origin.

UDC 681.3.01.002.235:002:65

Application of a Formal Word Processing Model to Improving Computer Administrative Document-Handling Technology

907G0057A Moscow KLASSIFIKATORY I DOKUMENTY in Russian No 12, Dec 89 pp 28-31

[Article by V. F. Aushev, All-Union Scientific Research and Design Institute of Computer-Engineering Technical Information]

[Text] Today administrative documents are generated based on graphics models of integrated document forms: standard forms (GOST 6.10.5-87) which are standardized designs for specific document forms in which raw factual data are accounted for.

Graphics models are oriented for important applications: elimination of document duplication as well as duplication of excess or unused document requests; generation of easy to use documents (both with respect to visual perceptibility and accuracy); and reduction of document preparation costs.

The graphics model has a substantially less effect on document word processing computer technology. In order to increase this effect it is advisable to generate an auxiliary formal document word processing model in the computer memory that is interrelated with an external graphics model.

The definition of the concept of text is significant in the design of a formal model. We will compare definitions of the concepts of the two most typical formal text models: the positional-alphabet model and the linguistic-graphic model.

In the positional-alphabet model the text is treated as a collection of positions where alphabet characters can be placed, and specific relations are then established for the position subsets. In general form text T is defined as

$$T = \langle S, \varphi \rangle$$

where S is the syntactical scheme, $S = \langle M; P_1, P_2, ..., P_n \rangle$ (here M is the set of text positions (places), $P_1, ..., P_n$ is the group of relations in set M); φ is the mapping of set M in character alphabet A.

Relation group $P_1, ..., P_n$ is selected to permit unambiguous definition for any linguistic object belonging to the entire set of administrative document texts. Ordinarily the relations in a syntactic scheme have the same names as the specific relations although they have a more complete set of properties.

The mapping φ for administrative documents is divided into two classes: individual mapping for the fixed and variable document text characters.

One characteristic feature of a positional-alphabet formal model is that the data description is closely related to the document form. In spite of its simplicity and broad utilization in automated management systems the positional-alphabet model has a significant drawback: The model can be used solely to retrieve and integrate simple text components of different documents (groups of words combinations and to some degree statements).

In the linguistic-graphics model the text is treated as a projection of the linguistic alphabet onto the graphical alphabet where independent relational systems exist for each alphabet.

The mathematical formulation of the concept of text (T) takes the following form:

$T = <S_l, S_g, \varphi>$,

where S_l is the linguistic text submodel;

$$S_l = \langle V;\ R_1^{S_1},\ R_2^{S_2},\ \ldots,\ R_n^{S_n} \rangle$$

(here V is the component alphabet of the linguistic text section;

$$R_1^{S_1},\ \ldots,\ R_n^{S_n}$$

is the group of semantic relations and properties defined in V where R_i is the predicate type and S_i is the predicate degree); S_g is the graphical text submodel, here

$$S_g = \langle M;\ G_1^{S_1},\ G_2^{S_2},\ \ldots,\ G_n^{S_n} \rangle$$

(here M is the alphabet of positions of the text graphical section);

$$G_1^{S_1},\ G_1^{S_2},\ \ldots,\ G_k^{S_k}$$

is the group of relations and properties defined in M, where G_l is the type of predicate while S_i is the predicate degree); φ is the mapping of the components V onto M.

The linguistic alphabet levels (V) include words, word combinations, groups of word combinations (words), correspondences of word combination (word) groups, statements, stanzas, fragments, and the linguistic portion of the text.

The graphics alphabet levels (V) include boxes, columns, column groups, and zones.

The components of each alphabet form embedded constructions.

The most characteristic linguistic relations include: succession, uniformity, composition, and direction. The graphics relations include the following types of relations; between box areas: merge and separate; between box lines; succession and justification.

One feature of the linguistic-graphical formal text model is that the description of the textual components is independent of document type. Such a degree of freedom complicates the description of the model although at the same time it makes it possible to retrieve and integrate textual components on any level throughout the entire collection of processed document texts.

The linguistic-graphical model was selected on the basis of this advantage as an auxiliary internal document text model in the computer memory. It is set up as a text data dictionary-handbook consisting of two types of data blocks: a common data block and a specific data block.

The general block will store descriptions relating to all types of document texts, particularly the axioms of the linguistic and graphical document structures. The specific blocks store descriptions on each type of document text (form), including a description of the linguistic alphabet mapping onto the graphical alphabet. The specific text data dictionary-handbook block consists of sections that include the general characteristics of the document form which consist of descriptions: document zones, column groups, individual columns, and boxes; fixed form requests; relational directories; word and word combination dictionaries; classifiers; request printing; composition of fragments, stanzas, statements, word combination groups, correspondence between word combination groups together with relational structures in the fragments, stanzas, statements, word combination groups, and the overall document.

The text data dictionary-handbook which is generated based on the linguistic formal model is used in many stages of computer word processing. This makes it possible to achieve a high degree of flexibility and continuity of data processing technology as well as semantic text integrity during its conversion in the data base while permitting retrieval and integration of the textual components on all levels throughout the entire set of administrative documents.

The problem of integrating the dictionary-handbooks remains as does the task of setting up an automated handling and distribution system in diskette form by a single information center in this field.

UDC 681.518.54

KONDITSIYA—A System of Modeling and Design of Digital Device Tests

907G0135A Kiev UPRAVLYAYUSHCHIYE SISTEMY I MASHINY in Russian No 2, Feb 90 pp 18-27

[Article by A. G. Birger, A. V. Boyarshinov, and Ye. S. Ryzhkov]

[Text] The current stage of development of digital technology is characterized by rapid growth in the degree of integration and functional complexity of the devices being developed, as well as ever larger use of LSI and VLSI circuits. The traditional "manual" methods of design under these circumstances lead to a drastic rise in the time and cost of the process, poorer product quality and reliability, and are becoming totally unacceptable when customized or semicustomized LSI and VLSI circuits are used in the hardware under development.

The only acceptable escape from this dilemma is to move to an automated design of hardware, making use of

computers. The necessary and most important stages of automated design of digital hardware are modeling and design of tests.

The system KONDITSIYA [1-6] may be used during these stages.

The KONDITSIYA system is implemented using the BESM-6 and Elbrus 1-K2 computers and is intended for modeling and automated design of tests for digital devices (DD) built from modern componentry. The KONDITSIYA system is oriented to direct working with the user (developer of the DD), which may take place either in interactive or batch mode. The system makes it possible to solve the following main problems:

—inspection and debugging of the source data descriptions;—verification (checking for correct design) of DD;—analysis of the tests for completeness;—design of tests of high completeness;—output of documentation.

Inspection and debugging of source data descriptions. The source data includes descriptions of the DD circuit, the test, and the burning of ROM and PLA microcircuits.

The system performs the following kinds of check: full syntactic; partial semantic and circuit-engineering check of DD description; full syntactic and semantic check of test description; full syntactic and semantic check of description of ROM and PLA burn; check for correspondence between descriptions of DD and test, and DD and burn.

The inspection makes it possible to discover and eliminate mistakes committed during the previous phases of the design process, and is done most efficiently in interactive mode. All messages regarding errors are immediately shown on the display screen. The resources present in the system allow prompt editing in interactive mode of the source data and a repeat inspection during the very same session. This iterative process is continued until all errors are corrected.

Verification of the DD. Verification is done by modeling the DD in a functional test—a "working" input sequence that is written by the developer of the DD. The verification essentially consists in comparing the behavior of the designed layout of the device against its functional algorithm.

The verification is done in different modes. In comparison mode, the user specifies in the functional test the values at both the inputs and the outputs of the DD. The system compares the output values obtained during the modeling with the values specified by the user and displays messages concerning mismatches. In the no comparison mode, the user specifies in the functional test the values only at the inputs of the DD and obtains for analysis the values at the outputs of the DD which are obtained by the modeling. Finally, in hybrid mode, the user may specify the values of the signals on certain outputs of the DD and not on others. The results of the modeling, at the option of the user, may be put out to alphanumeric printer and/or to the display screen in a form convenient for analysis. We shall enumerate the various kinds of output of results.

First, in the comparison mode, the alphanumeric printer and/or display screen outputs the numbers of the test checks and the DD outputs at which mismatches are discovered.

Second, the alphanumeric printer may output a table of stable signal values obtained as a result of the modeling in all test checks at all internal points of the DD.

Third, the KONDITSIYA system realizes a "multichannel probe," making possible selective output on printer and/or display screen of both stable signal values and time diagrams for any given user-designated contacts of the microcircuits and connectors in any given range of test checks.

Fourth, the system outputs a list of circuits in which the signals did not take on all possible values during the time of the test.

If the signal values on the DD outputs obtained during the modeling do not agree with the values expected by the user, it is necessary to scrutinize the information obtained. The following conclusions may be reached as a result of this:

—improperly designed DD;—improperly specified burn of the ROM and/or PLA microcircuits;—incorrectly developed test;—the modeling method used is not adequate for the particular DD.

In the first three cases, the respective source data should be corrected by means of the interactive features of the KONDITSIYA system, after which the verification is repeated. In the fourth case, it is necessary either to modify the description of the DD in order to improve the adequacy of the modeling (see below) or to change the modeling method.

The KONDITSIYA system implements three methods of modeling in a four-character alphabet (0, 1, X, Z), where the character X corresponds to an undetermined (unknown) signal value, while Z is the character of the third (high impedance) state.

The modeling methods implemented in the system are distinguished by the fact that the model of the DD, in addition to the logic gates, may contain complex multiple-output components with memory (macrocomponents). Use of the macrocomponents improves the adequacy and speed of the modeling and reduces the computer memory volume required. The modeling methods differ in having different provisions for the time delays in the components and the connections.

The four-place modeling with use of delays of components and connections, which is based on the ideas of Eichelberger [7], makes it possible to identify all contest conditions in the circuit that may occur in the given test. The advantages of the method include its high speed and relatively modest computer memory requirement. The modeling is realized as a parallel compilation, so that it is possible to model simultaneously either 48 test checks in a properly working DD or one properly working DD

and 47 faulty DD's in one test check. A major disadvantage of this method is the detection of nonexistent ("superfluous") contest conditions in certain cases.

The system KONDITSIYA has resources for broadening the area of application of the modeling with arbitrary time delays, if the "superfluous" contest conditions are due to the presence in the DD of standard designed subcircuits. Standard designed subcircuits often make use of the circumstance that the delays in the connections of real world DD's are much shorter than the delays of the components. Given this presumption, multiple-bit counters (registers) are built, e.g., from counters (registers) of shorter word length.

The library of components of the KONDITSIYA system [5] includes modified models of the microcircuits used to construct standard subcircuits (counters, registers, single pulse generators, etc.).

Use of the modified models ensures an adequate modeling of the standard subcircuits. If the DD contains similar standard circuit engineering solutions, then it is possible to achieve adequate modeling results by replacing the respective microcircuits in the source description of the DD with the modified microcircuits.

The four-place modeling with unit (equal) delays is carried out on the assumption that all logic components in the circuit have an identical delay time, and the delay times in the connections are equal to zero. Allowance for the spread in delay times in the circuit components, which is necessary in order to discover contest conditions, is accomplished in part by "blurring" the times of switching of the signals on the input pins of the circuit. This modeling, being an extension of the method of Δ-ternary modeling [8] to a four-character alphabet, is also realized as a parallel compiling and makes it possible to obtain adequate results for a large class of DD, which is the reason for its many practical applications. However, this method does not guarantee that all contest conditions in the DD will be detected. Moreover, if the time delays of the DD components differ significantly from each other and if this plays an important part in the functioning, the results of the modeling may not coincide with the expected results. Such inadequacy of the modeling may be eliminated by introducing fictitious delay elements in the description of the DD.

Four-place modeling with nominal delays of the components makes it possible to obtain adequate results in the modeling of DD's, the functioning of which depends on a "game of delays." This is realized as a single-event interpretive modeling and is mainly oriented to deal with a properly working DD. Unlike the two previous methods, this modeling method makes it possible to determine the specific times of actuation of both the DD in its entirety and its individual subcircuits. A major advantage of the method is the fact that it is able to handle devices of much larger dimension than the two previous methods, since modeling parallelism is abandoned. The drawback of the method of modeling with nominal delays is that it is only able to identify a small number of the contest conditions in the DD.

Practical use of the KONDITSIYA system shows that the most effective is an integrated use of the different modeling methods. To verify the DD, the modeling with arbitrary delays is first employed. If the circuit is designed such that it logical behavior in this test is not supposed to depend on the relationship among the time delays in the components and connections, the modeling will yield absolutely correct results. But if the logical behavior of the circuit may depend on the relationship among the delay times, then indeterminate values may be obtained on the respective lines of the DD. Certain of these may be "superfluous."

In this stage of the modeling with arbitrary delay times, the designer of the DD should correct the source data to make the signal values at all output pins of the DD correspond to the anticipated values, with the possible exception of those outputs which are in an indeterminate state on account of false (in the opinion of the designer) contest conditions. In the latter case, one should begin a modeling either with unit or nominal delays, depending on the design philosophy of the DD being handled, until full correspondence is achieved between the modeling results and the anticipated results.

During the verification, the test is modeled either in its entirety or in arbitrary chunks, as the user desires; in particular, a step-by-step modeling mode is possible. If the session time elapses in the course of the modeling, the count is automatically broken off. The user may write the interim results in the archive and continue the modeling in the next counting session from the point of the interruption.

The quality of the verification depends on how thoroughly the functional test encompasses the operating conditions of the DD. At a minimum, the functional test should provide for "all switching events" (i.e., each permissible signal value should be set up at least once in each circuit of the DD). After the user has brought about full correspondence between the modeling results and the anticipated results, a list of the circuits in which the signal values were not switched to all permissible states may be put out on printer and on the screen. Using this list, the test should be supplemented to achieve a high quality of verification.

Verification makes it possible to identify and correct a considerable number of the mistakes in design of the DD, the test, and the ROM and PLA burn.

Design of highly complete tests. The prime quality characteristic of a verifying test is completeness (the percentage of errors found). In the KONDITSIYA system, the completeness of the test is defined in terms of the set of single constant malfunctions of type "identical to 0" and "identical to 1" at the data outputs of the microcircuits and connectors. The analysis is performed by modeling a properly working DD and a DD with each of the malfunctions in question. It is considered that the

malfunction has been checked if, in at least one test check on one of the output pins the signal value in the properly working DD is equal to 1 (0), and in the malfunctioning 0 (1). The outcome of the test completeness analysis is a list of malfunctions not verified in the given test in the form of a diagnostic table.

The design of a highly complete test begins with analysis of a functional test that has been worked out in the verification stage. If this test provides for all switching events, then it usually also has sufficiently high completeness. Moreover, the condition "all switching events" must be fulfilled to obtain a test of one hundred percent completeness. However, this condition is not sufficient, and thus it may be necessary to supplement the test.

The test may be supplemented automatically or manually. The KONDITSIYA system realizes two methods of automatic test synthesis. First, a random search is made, making it possible to quickly construct a test which checks for "easily detected" malfunctions. The efficacy of the random search then rapidly declines. After this, a guided test synthesis algorithm is brought into action, which requires large expenditure of machine time, but yields highly complete tests in a number of cases.

The guided algorithm is based on the idea of activation of pathways and employs a computation of D-cubes [9]. The centerpiece of the algorithm are the procedures of D-movement, support, and implication for the individual microcomponents, the models of which are structured automatons. The iterative asynchronous model of the DD, which is used to synthesize the test, takes into consideration not only persistent, but also transient states of the DD. In order to reduce the sorting, use is made of parameters of comparability and observability, which are automatically computed before the algorithm is executed. However, this method does not guarantee absence of contest conditions in the test that is constructed. Each sequence of "test candidates" constructed by means of the guided algorithm to check for the next kind of malfunction is modeled in order to determine its checking properties.

It should be mentioned that the automatically constructed test can only serve as a supplement to the functional test of the designer, but will not replace it, since only a qualitatively conducted verification in the functional test of the designer guarantees that the intention of the creator of the DD will be adequate to the resulting circuit of the device.

If the automatic test synthesis does not ensure completeness that is satisfactory to the user, it is possible to supplement the test manually. The "manual" test is joined to the previously constructed test and put through a simulation. If the simulation demonstrates the usefulness of the newly incorporated checks, i.e., if they do not check for already checked malfunctions, then these checks are left in the test. Otherwise, they should be replaced with different ones. This process may be repeated several times until the necessary test completeness is achieved.

If the test is sufficiently complete, the diagnostic information obtained as a result of the simulation can be effectively used to look for defects when checking the DD.

Output of documentation. The chief outcome of the running of the KONDITSIYA system is a test that ensures high completeness of checking for malfunctions during manufacture and use of the DD. The system outputs punched control tapes for test inspection units of type UTK.

For DD's which contain ROM and/or PLA microcircuits, the running of the system also produces a debugged description of the burn for these microcircuits. The system outputs punched control tapes for the automatic burning machines.

Furthermore, the system prints out the following test documentation, to be used in the course of the design and debugging of the DD: printout of descriptions of the DD, the test, and the burn process; a table of circuits; a table of the test; a table of signal values at internal points of the circuit; time diagrams; a list of groups of indistinguishable malfunctions; a diagnostic table; and a list of unchecked malfunctions.

The system also provides accompanying information (messages and supplementary printouts characterizing the DD being worked on and the problem solving process) and emergency messages (when mistakes are found in the source data).

When operating in interactive mode, all information needed by the user is shown on the display screen.

Basic characteristics of the devices handled by the system. The KONDITSIYA system can handle digital storage cells, digital array LSI (DA LSI) circuits, and other DD's amenable to a test inspection in asynchronous operating mode.

The DD's may include microcircuits (components) of small or medium degree integration, as well as LSI circuits (in limited amount). The operating possibilities of the system depend not only on the number of microcircuits in the device, but also on the types and recurrence of the microcircuits, and therefore it is not possible to give the utmost permitted number of microcircuits in the DD. Tentatively, for DD's built with components of small (medium) degree integration, the upper bound is an equivalent complexity of 4,000 (10,000) gates.

The KONDITSIYA system also has format restrictions:

—the DD may not contain more than 4,095 circuits;—the total number of logic components and external contacts of the DD should not exceed 4,095;—the number of external contacts of the DD should not exceed 400;—the overall number of internal variables

in the DD model should not exceed 4,095 (not counting internal variables belonging to the ROM, which are handled separately). Furthermore, use of "abbreviated " RAM models makes it possible to handle DD's which contain a RAM of any desired capacity (see below).

The system enables automatic processing of DD's which contain resistors, capacitors, components with tristable and bidirectional pins, bus structures, and components with open collector. This generally obviates manual touching up and editing of the DD description prior to the modeling.

The component description library (CDL). The library includes microcircuits of small, medium, and large degree of integration (series 109, 130, 133, 134, 136, 169, 185, 514, 530, 533, 537, 541, 556, 559, 564, 571, 580, 585, 1509, 1517, 1531, 1533, 1802, 1804). The CDL also includes basic components of DA LSI circuits, resistor assemblies, connectors, analog radio components, and so on. Furthermore, the CDL includes conventional components that are used, in particular, in the modifications of the DD descriptions. At present, the library contains more than 600 components.

The description of a component contains information as to the number and purpose of the contacts of the component, its functional algorithm, the delay times, and certain other information. The functional algorithm may be given either by a structural model or equivalent circuit, consisting of components previously incorporated into the CDL, or by a functional model by means of branch and output functions [1, 3]. The CDL is built on the hierarchical principle—the components that are structurally described may themselves be used in other structural models as components of the equivalent circuit, and the hierarchy has unlimited levels. The system has sophisticated development tools for entry, checking, and certification of the component models.

The CDL is a common use library and models are entered into it only by the designers of the system. However, the users have the option of entering independently developed models into personal libraries (which is chiefly useful when the required models are lacking in the CDL).

The development of functional models for the components, especially for complicated components of medium and large scale integration, is an arduous nonformal procedure. Such models must obey stringent requirements, since they must be universal, and the very same models must be used by all three simulation methods and by the guided test synthesis algorithm. Therefore, the users are currently restricted to development of structural models and entering of same into their personal libraries.

A special place in the CDL is held by the ROM and PLA microcircuits, the functional description of which requires knowledge of the information that is "sewn into" in each particular microcircuit. Accordingly, the source data of the system should contain a description file for the ROM and PLA burn, if the DD contains the respective microcircuits. The information in this file is specified individually for each microcircuit: for the ROM, in the form of a sequence of codes in order of increasing addresses; and for the PLA, in the form of Boolean functions. The models of the ROM and PLA microcircuits are synthesized as needed prior to the simulation, starting with the library information and the description of the burn.

Let us consider the handling of random access memories (RAM) in the KONDITSIYA system. As mentioned above, the total number of internal variables in the DD model cannot be greater than 4095. In practice, this is an important restriction if the DD being handled contains RAM microcircuits of large capacity. In order to overcome this restriction, the system provides for construction of a DD model by using "abbreviated" RAM models, which makes it possible to simulate DD's containing a RAM of any desired capacity.

The abbreviated RAM model corresponds to the actual RAM, in which several inputs responsible for the high-order address bits have been "torn off." The number of address inputs that must be torn off for the RAM to fit in the computer memory is determined by the software and presented to the user as a message. Thus, when simulating a RAM having n address inputs and a corresponding memory volume of 2^n bits, only the k ($k<n$) low-order address inputs are considered, while the signal values at the other n-k inputs are ignored. In this case, 2^k memory bits ($2^k<<2^n$) are enough to store the state of the microcircuit. The values corresponding to the addresses of the actual microcircuit that are multiples of 2^k will be saved at the identical variables of the abbreviated model, which requires a special approach to forming the functional test for the verification.

Thus, for example, if the designer's test uses only the 2^k low-order addresses of the given RAM, then such test will be correctly simulated by the KONDITSIYA system. However, certain operating modes of the DD may go unchecked. Furthermore, the signal values on certain lines will remain unchanged. All of this will impair the completeness of the test.

There are techniques for "sounding" all of the addresses of the actual RAM. For example, the test may be organized such that after the information is written into the RAM at a certain address this same information is read at the same address. Such test also will be correctly simulated, although the write and read addresses in the actual RAM and those in the abbreviated RAM model may not coincide.

It is also possible to write information consecutively at various addresses of the RAM, and only then read it. In this case, however, one must not permit writing at different addresses of the actual RAM, which turn into a single address of the RAM model, so that the useful information is not erased.

By constructing a test in view of the foregoing recommendations, it is possible to access any given addresses of the RAM and provide for the switching of signals on all lines of the DD. However, the KONDITSIYA system will reduce the actual completeness of the test in this case. For example, malfunctions at the detached address inputs will always be announced as unchecked, even though they might have been checked in the actual DD.

Implementation and operation. The KONDITSIYA system is an evolving package of programs written in algorithmic language LYaPAS-M. In order for the system to work, it requires the operating system DISPAK, the programming system PROLOG, a working storage of 32 pages, and 120 tracks on magnetic drums. To support the interactive mode, a display of VT-340 or RIN-609 type is needed. The programs and the library of component descriptions of the KONDITSIYA system in its minimal configuration occupy 4408 zones on magnetic disk (data as of late 1987).

The KONDITSIYA system may be used either as a subsystem in the start-to-finish process of design of digital cells and LSI circuits or by itself. The system has been in use since 1983 and has been adopted by a number of organizations. During this time, the system was used to design hundreds of DD. The system is set up on the BESM-6 or Elbrus 1-K2 computers, the identical problem on the Elbrus computer running 3-4 times faster than on the BESM-6.

The programs of the KONDITSIYA system are independent modules which are kept in translated form in the libraries of program modules. Exchange of information between programs is done through a unified database, which is set up as a system on magnetic drums. The system has its own archives on magnetic disks and/or tapes, which makes possible storage of the source, interim, and output information. Use of these archives makes it possible to break up the operating cycle of the system into several sessions, to switch from interactive to batch mode and back, to halt the running of the system at any given point, and then resume from the interruption point in another computing session, and to repeat the computation from any given terminated program.

Control of the running of the system is done in the language of commands. The user specifies the entire necessary sequence of commands either all at once or in several batches. In batch mode, the results of execution of the commands are printed out on alphanumeric printer, and the computing session is automatically terminated after executing the last command. In interactive mode, the results of execution of the commands are shown on the display screen, and if the user desires they are also printed out; after execution of the last command, the computing session is not terminated, but instead control is transferred to the user.

The KONDITSIYA system makes it possible to enter information from terminal, punched tape, and punched cards, and to save it in private archives. Facilities have been developed to support the exchange of information between archive of the KONDITSIYA system and the archives of other systems (in particular, the general-purpose systems PULT and KRAB). The interactive editor of the system allows viewing and editing of any given texts (in particular, a listing), access to information in different archives, and a number of other options. There is also a specialized interactive text editor enabling insertion, emendation, and deletion of test checks, examination of a test by checks and by channels, and a number of other operations.

The KONDITSIYA system is accompanied by more than 400 pages of user instructions. Some of these instructions are written on magnetic disk in the form of a "system tutorial" and come with the system. If the user has difficulty in working with the system in interactive mode, the information of interest (such as a list of system commands, possible alternatives for continuing the operation, a description of any given command) may be called up on the display screen and studied as needed. In addition, a small example (i.e., source descriptions of a DD and test) is incorporated into the system. Running this example, which is accompanied by detailed commentary in the instructions, allows rapid assimilation of the technique of using the system.

Bibliography

1. "KONDITSIYA, an automated system of inspection and diagnostics of digital cells," A. G. Birger, A. V. Boyarshinov, M. Yu. Vinokur, et al., ELEKTRON. MODELIROVANIYE, No 4, 1980, pp 58-63.

2. "An automated system for construction of tests for digital cells of heightened degree of integration," A. G. Birger, A. V. Boyarshinov, M. Yu. Vinokur, et al., OBMEN OPYTOM V RADIOPROMYSHLENNOSTI, Issue 4, 1983, pp 4-9.

3. Boyarshinov, A. V., Ryzhkov, Ye. S., "Mathematical models of logic components in an automated test construction system," Ibid., Issue 11, pp 43-46.

4. Birger, A. G., Boyarshinov, A. V., Ryzhkov, Ye. S., "Formation of mathematical models of a digital device in a test construction system," Ibid., Issue 4, pp 43-48.

5. "The library of component descriptions of a system for analysis and synthesis of tests for digital cells," A. G. Birger, A. V. Boyarshinov, M. Yu. Vinokur, et al., Ibid., Issue 11, pp 36-39.

6. Birger, A. G., Boyarshinov, A. V., "Organization of the information exchange in an automated test construction system for digital devices," Ibid., Issue 11, 1984, pp 18-21.

7. Eichelberger, E. B., "Hazard detection in combinational and sequential switching circuits," IBM J. RES. AND DEVELOP. Vol 9, No 2, 1965, pp 90-99.

8. "Avtomatizirovannoye proyektirovaniye tsifrovykh ustroystv" [Automated design of digital devices], ed. by S. S. Badulin, Radio i svyaz, M., 1981, 240 pp.

9. Roth, J. P., Bouricius, W. G., Schneider, P. R., "Programmed algorithms to compute tests to detect and distinguish between failures in logic circuits," IEEE TRANS. Vol 16, No 5, 1967, pp 567-580.

22161
30
NTIS
ATTN: PROCESS 103
5285 PORT ROYAL RD
SPRINGFIELD, VA
22161

Foreign Broadcast Information Service (FBIS) and Joint Publications Research Service (JPRS) publications contain political, military, economic, environmental, and sociological news, commentary, and other information, as well as scientific and technical data and reports. All information has been obtained from foreign radio and television broadcasts, news agency transmissions, newspapers, books, and periodicals. Items generally are processed from the first or best available sources. It should not be inferred that they have been disseminated only in the medium, in the language, or to the area indicated. Items from foreign language sources are translated; those from English-language sources are transcribed. Except for excluding certain diacritics, FBIS renders personal and place-names in accordance with the romanization systems approved for U.S. Government publications by the U.S. Board of Geographic Names.

Headlines, editorial reports, and material enclosed in brackets [] are supplied by FBIS/JPRS. Processing indicators such as [Text] or [Excerpts] in the first line of each item indicate how the information was processed from the original. Unfamiliar names rendered phonetically are enclosed in parentheses. Words or names preceded by a question mark and enclosed in parentheses were not clear from the original source but have been supplied as appropriate to the context. Other unattributed parenthetical notes within the body of an item originate with the source. Times within items are as given by the source. Passages in boldface or italics are as published.

SUBSCRIPTION/PROCUREMENT INFORMATION

The FBIS DAILY REPORT contains current news and information and is published Monday through Friday in eight volumes: China, East Europe, Soviet Union, East Asia, Near East & South Asia, Sub-Saharan Africa, Latin America, and West Europe. Supplements to the DAILY REPORTs may also be available periodically and will be distributed to regular DAILY REPORT subscribers. JPRS publications, which include approximately 50 regional, worldwide, and topical reports, generally contain less time-sensitive information and are published periodically.

Current DAILY REPORTs and JPRS publications are listed in *Government Reports Announcements* issued semimonthly by the National Technical Information Service (NTIS), 5285 Port Royal Road, Springfield, Virginia 22161 and the *Monthly Catalog of U.S. Government Publications* issued by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

The public may subscribe to either hardcover or microfiche versions of the DAILY REPORTs and JPRS publications through NTIS at the above address or by calling (703) 487-4630. Subscription rates will be provided by NTIS upon request. Subscriptions are available outside the United States from NTIS or appointed foreign dealers. New subscribers should expect a 30-day delay in receipt of the first issue.

U.S. Government offices may obtain subscriptions to the DAILY REPORTs or JPRS publications (hardcover or microfiche) at no charge through their sponsoring organizations. For additional information or assistance, call FBIS, (202) 338-6735,or write to P.O. Box 2604, Washington, D.C. 20013. Department of Defense consumers are required to submit requests through appropriate command validation channels to DIA, RTS-2C, Washington, D.C. 20301. (Telephone: (202) 373-3771, Autovon: 243-3771.)

Back issues or single copies of the DAILY REPORTs and JPRS publications are not available. Both the DAILY REPORTs and the JPRS publications are on file for public reference at the Library of Congress and at many Federal Depository Libraries. Reference copies may also be seen at many public and university libraries throughout the United States.